AD-A080 767 HONEYWELL SYSTEMS AND RESEARCH CENTER MINNEAPOLIS MN F/G 17/2
SYSTEM CONTROL FOR THE TRANSITIONAL DCS.(U)
MAY 79 F C ANNAND, D E DOTY, R V KRZYZANOWSKI DCA100-78-C-0017
UNCLASSIFIED TR-3 SBIE-AD-E100 328 NL

1 OF 4
AD
A080767

AD-E100 328

(2) LEVEL III

A080766

Honeywell

SYSTEM CONTROL FOR THE TRANSITIONAL DCS

TECHNICAL REPORT NUMBER 3

F. C. Annand
D. E. Doty
R. V. Krzyzanowski
R. A. Schlicht

DEFENSE COMMUNICATIONS AGENCY
DEFENSE COMMUNICATIONS ENGINEERING CENTER
1860 Wiehle Avenue
Reston, VA 22090

Under Contract DCA 100-78-C0017
May 1979

SYSTEMS & RESEARCH CENTER

AEROSPACE & DEFENSE GROUP

2700 Ridgway Parkway
Minneapolis, MN 55413

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

UNCLASSIFIED May 1979

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
Three

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
(6) System Control for the Transitional DCS. ~~Technical~~ Report Number 3

5. TYPE OF REPORT & PERIOD COVERED
Interim Report
Jan 1978 - Sep 1979

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
(10) F. C. Annand, R. V. Krzyzanowski, D. E. Doty, R. A. Schlicht

8. CONTRACT OR GRANT NUMBER(s)
(15) DCA 100-78-C-0017

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Honeywell Systems and Research Center
Aerospace and Defense Group
2700 Ridway Parkway, Minneapolis, MN 55413

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Defense Communications Agency DCEC-R310
1860 Wiehle Avenue
Reston, VA 22090

12. REPORT DATE
(11) May 1979

13. NUMBER OF PAGES
300

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
(12) 335

15. SECURITY CLASS. (of this report)
UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Approved for Public Release - Distribution Unlimited

(18) SCTL (19) AD-E100 328

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)
(14) TR-3

18. SUPPLEMENTARY NOTES
(9) Interim rept. Jan 78-Sep 79

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

DCS	Data Base	AUTOVON Traffic Control
Altrouting	Data Base searches	Network Control
Altrouting Algorithms	Scenarios	
Patching	Benefits Analysis	
Automated Patching	Transmission Control	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report, the third of four reports required under two tasks, discusses system level control actions and algorithms for their implementation. The report is directed at the projected DCS of the mid 1980's. It presents algorithms for automatic searches through the connectivity data bases developed in report #2 for determining potential circuit and trunk restoration routes. Other algorithms presented are aimed at adjusting the routing in the AUTOVON Network.

DD FORM 1473, 1 JAN 73 — EDITION OF 1 NOV 65 IS OBSOLETE

402349

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

SYSTEM CONTROL FOR THE TRANSITIONAL DCS

TECHNICAL REPORT NUMBER 3

F. C. Annand
D. E. Doty
R. V. Krzyzanowski
R. A. Schlicht

DEFENSE COMMUNICATIONS AGENCY
DEFENSE COMMUNICATIONS ENGINEERING CENTER
1860 Wiehle Avenue
Reston, VA 22090

Under Contract DCA 100-78-C0017

May 1979

HONEYWELL INC.
Aerospace & Defense Group
Systems & Research Center
2700 Ridgway Parkway
Minneapolis, MN 55413

CONTENTS

CONTENTS

CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION I

INTRODUCTION AND SUMMARY

1.0 PURPOSE OF THE REPORT

This report describes system level control actions appropriate to the Transitional Defense Communications System (DCS) of the mid to late 1980's. It is the third technical report of the System Control for the Transitional DCS Study. The purpose of this study is to define the functional characteristics of an automated system which will provide the information collection and utilization capabilities needed by theatre/ACOC level in the performance of its real-time mission, and the relationship of this system to lower level system control subsystems. The peace-time mission of DCS is the primary focus of the study, although the system defined would also be useful in certain hostile situations.

The first report in this study established the assumed characteristics of the DCS of the mid-1980's, and the second report discussed the information collection/display recommendations for system control. This report discusses the utilization of that information for control. The fourth report will estimate the size and cost of these utilization capabilities, and will relate these costs to the benefits obtained.

The remainder of this section will summarize the work of the first and second reports, and provide an overview of this report. Section II discusses transmission system control, and Section III discusses AUTOVON control.

1.1 BASELINE ASSUMPTIONS

The study focuses on the European area of the DCS, as it is anticipated to exist in the mid to late 1980's. The European area was chosen because it is as complex as any other segment of the DCS and contains examples of every type of subsystem used in the DCS. It is therefore reflective of mission objectives world-wide, and the results can be directly extended to other theatres. The European DCS as defined in the first report (reference 1) was assumed to consist of:

- o A digital European backbone using microwave radios and multiplex equipment compatible wtih DRAMA specifications and digital tropo scatter radios.

- o The ATEC system, as specified by the FSD ATEC 10000 specification, fully deployed for monitoring, fault isolation, and reporting of transmission system status.

- The Defense Satellite Communication System using the DSCS III satellite, under the control of equipment specified by the DSCS control segment specifications.

- An AUTODIN II system with three packet switching nodes, identical to those being developed for use in CONUS, replacing existing European AUTODIN switches.

- An integrated AUTOVON/AUTOSEVOCOM II system using AN/TTC-39 switches, to replace the 490L AUTOVON switches, and SB-3865 secure switchboards.

Congressional action during the study has made this last assumption obsolete. Therefore, for this report, and for the fourth report which will follow, the common user circuit switched network is assumed to be either the 490L network upgraded with the rapid access maintenance monitor (RAMM), enhanced routing capability and common channel signalling, or some other replacement switch built basically to commercial standards. Secure voice is assumed to be provided over this network in a transparent manner.

1.2 SYSTEM CONTROL PHILOSOPHY

The basic philosophy of system control for this study is that control will take place at the lowest level in the control hierarchy which can feasibly perform it. As problems occur which cannot be solved at any level, they "bubble-up" to higher levels.

The capability for real-time control is already being developed for the transmission system at levels below Theatre. The switched networks have essentially a two-level hierarchy consisting of the switches themselves and the Theatre level control. This study therefore concentrates on the system control functions which are performed at the Theatre level. These functions can be described as network connectivity control, AUTOVON control, and AUTODIN control.

Network connectivity control functions at the Theatre level can be considered a direct extension of today's policies. In this case, Theatre level activity will be initiated by a request from sector level personnel for assistance in handling an outage. Typically, this would occur when multiple failures have rendered restoral plans obsolete, or when plans are found to be out of date or missing. Theatre level can then create a restoral plan on-line with the assistance of its computer system, and provide sector with a recommended restoral action.

In order to provide this assistance, the Theatre facility must have accurate, current knowledge of the configuration and status of all transmission resources. It must also have searching algorithms capable of generating a good restoral plan in a reasonable amount of time.

Given these capabilities, it is reasonable to consider an alternative policy of operations. Upon the occurence of any status change, the Theatre level processor can determine if a preplanned altroute is applicable to the problem. If such a plan exists, no further action is required except to update the data base when lower levels report that the plan has been implemented. If no plan exists applicable to the current situation, the Theatre level computer could create a plan on-line and pass it down as a network control directive. The lower levels could then either implement or take exception to the directive.

This method of operation, with decisions made at the higher levels and passed down as directives, leads naturally to a more automatic network control system using remote control patching devices. At this stage in the evolution of the control structure, status changes (faults) detected by ATEC will trigger the automatic creation of a restoral plan. A controller at the Theatre level will be alerted that a restoral plan needs review. After reviewing the restoral plan on a CRT display, the controller can either direct its implementation, make specific changes to the plan, or give the computer directives about the plan which would cause the computer to generate a new plan. An example of a directive would be "don't preempt circuit XXXX1234", because the controller knows that the user of that circuit has more need than priority, or "don't use link TOXXX unless absolutely necessary", because the controller knows that weather conditions are poor for tropospheric transmission.

When the plan is finally approved and the controller puts in the implement command, the computer would broadcast the list of directives to each affected station. At the station, the messages would be received by automated patching devices which would implement the directed patches.

This policy of operation is a large evolutionary distance from current operations. However, it will provide better service to the user, particularly in the area of critical subscriber connectivity. We have said that these functions are Theatre level functions. However, there is no reason that these functions could not be performed at the Sector level. Each Sector, by virtue of the intersector communications system, has full visibility of the entire Theatre. Thus, the network connectivity control function could actually be performed at Sector, or the Theatre level control could be backed up by Sector.

The second technical report (Reference 2) provided recommendations for the data collection and display function to support the first step in enhnacing network connectivity control. Parameters which are needed from the communication subsystems in order to support high level decision making were selected. Communication paths for these parameters have been defined. A

data base schema was developed to contain the status information at Theatre, and a complete set of displays for providing access into the data base was described. A summary of the data base schema, and the changes to it required in order to also support automated restoral plan generation, is provided as Appendix A to this report. Appendix B describes the communication paths and the message formats recommended for network connectivity control.

The AUTODIN II packet switching network has, as a part of the network system design, an extensive system control capability. Part of this capability is inherent to the system concept, such as the adaptive routing procedure, and part of this capability is specifically added in the form of a Network Control Center (NCC). The NCC provides management with the information collection and display capabilities necessary to support real-time decision making as well as long-term engineering functions. The NCC also has the capability to regulate the operation of the network. Because this extensive control capability has already been designed for the CONUS portion of the network, it was recommended that an identical copy be procurred for controlling the European subnetwork. This recommendation is continued for this portion of the study, i.e., the NCC has sufficient capabilities when combined with the inherent qualities of the AUTODIN II design that no additional facilities need be added for AUTODIN II control.

For AUTOVON control using the TTC-39 switch, the second report recommended that all of the parameters currently reported on the system control subchannel be collected at the Theatre level facility. The use of traffic parameters was basically long-term engineering, since it is necessary to smooth them for long periods of time. Equipment status parameters were used to make decisions about routing table changes and restrictive traffic controls needed for smooth operation of the network.

Basically, the same parameters are needed regardless of the details of switch construction. Traffic data is needed for long-term engineering, and equipment status is needed to make real-time control decisions. Given complete freedom to specify the characteristics of the circuit switch, the amount of control involvement can be minimized. However, visibility of system operation is still needed so that unanticipated stresses requiring manual override can be handled.

1.3 REPORT OVERVIEW

Section II is a discussion of network connectivity control. Current procedures, directives, and circulars relating to the management of network connectivity are reviewed. It is shown that because of the preponderance of 64Kb circuits, it is possible to consider automatic altroute searching algorithms. Algorithms are presented for automatically creating restoral plans on line, both in the baseline system and with the

availability of remotely controlled channel reconfiguration units. The impact of these algorithms on restoral time and circuit availability is demonstrated, with and without remote configuration, for certain representative situations. Finally, a discussion of how modern DBMS concepts can be used to simplify the implementation and increase the performance of the algorithms is presented.

Section III discusses control of the AUTOVON system. The discussion is oriented towards control of circuit switched networks in general. Literature describing the evolution of traffic control is referenced, which shows that traffic control is a basically undesirable action which is required to compensate for the finite switching capability of the circuit switches. Procedures used by the Bell System and other commercial networks are presented. These topics lead naturally to recommendations for the 490L system. These recommendations are heavily influenced by the improvement programs currently under way or planned. The primary recommendation is that some form of adaptive routing can be implemented to maximize the effective network connectivity. No new traffic control mechanisms are recommended, although better information for the controllers in determining when to apply controls is recommended. A recommendation is made that when improved routine control and common channel siganlling is installed, it be over specified such that traffic controls are not needed. Also, new routing control should support rehoming of critical subscribers such that their telephone number does not change during rehoming.

SECTION II

TRANSMISSION SYSTEM CONTROL

2.0 INTRODUCTION

As a major portion of the control activity proposed for the transitional DCS in the 1980's, resource allocation (i.e., "altrouting") of the DCS transmission facilities is addressed in these sections. Discussion will begin with the present day requirements for such control action and follow through to specific algorithms that carry out a rational program of resource allocation. Finally, an analysis of the benefits of such control actions will be estimated.

Before we enter into the details of the control, let us first set the stage we will be working on. The DCS is a communications network designed to transmit and distribute a variety of circuits (both dedicated and switched, common-user). In the 1980's, the backbone of the network and large parts of its distribution links will be digital time-division multiplexed transmission facilities. Circuits will pass through either two or three levels of multiplexing to the transmission level. Over their route, they may stay at this highest level of multiplexing or they may move down one or two levels in order to permit reconfiguring them into new multiplex groups to reenter the transmission environment. At these points of multiplex level changes, the electrical compatibility is well defined so as to allow the maximum possibility for reconfiguring circuits without worry of level incompatibility. In fact, it is this uniformity of circuit appearance that makes the possibility of examining altrouting in an automated manner a reality.

When failure or serious degradation of equipment occurs along the reroute of a circuit, attempts are made to find spare equipment over a new or partially new route in order to restore the service. In the event that no spare facilities are available to complete an altrouting of the service, the relative importance of the service is compared to the services occupying a possible altroute in order to determine whether those facilities could be temporarily pre-empted from their normal users in order to altroute. The basis for this decision will be a catalog of restoral priorities (RP's) assigned to each service to rate its importance relative to other users services. This assignment is crucial to either a human or automated decision process. Its validity and maintenance will determine whether an altrouting algorithm will succeed in providing a high grade of service to critical service users or fail and promote arbitrary pre-emption and over-all service degradation. The establishment of these categories is an important administrative task that will not be addressed, but is no less crucial to the success to this work than the algorithms themselves.

2.1 THE CURRENT ALTROUTING PROCEDURE

2.1.1 Procedures Called Out in Current Documentation

A study of the current procedures as they relate to altrouting is made first for two reasons. First, this information provides a basis to compare the algorithms to be developed to current procedures in order to determine benefits of the algorithms. Secondly, the algorithms developed should follow existing DCA operational directives as they relate to altrouting.

The primary responsibility of Technical Control Facilities (TCF's) relating to altrouting is spelled out in DCAC 310-70-1 (Ref. 4). The procedure to altroute a failed service is to start the process at the TCF that first discovers the failure. This activity should use the NCS RP system to determine how to pre-empt circuits when that option must be exercised. To aid in the altrouting, the TCF's should have some action outline to be followed in the event of a failure.

Those TCF's that are designated as Facility Control Offices (FCO's) are assigned to the responsibility of restoral over the wideband transmission facilities they control. They must have altrouting plans approved by the appropriate DOCC elements and be ready to implement them upon requests from TCF's that are coordinating the restoral action. The current transmission multiplexer maps and trunk routings should be at these sites to assist in the altrouting action.

These directives are all that are given. Specific instructions on coordination of the altrouting is not mentioned in this document.

DCA Circular 310-55-1 (Ref. 5) is more specific in detailing the altrouting responsibilities and procedures. The line of responsibility begins at the station level and procedes upward in the DCS command structure until a level that can solve the problem is reached. The only exception to this is for circuits connected to other areas. In this case, altrouting responsibility starts at ACOC.

Restoral plans should be developed by the DCS command hierarchy in cooperation with the local CINC's and O&M's for as much as the DCS facilities as they control and can manage to altroute. Circuits that have no altroutes will be identified.

The DCS station which first recognizes a failure will undertake fault isolation and altrouting actions. When that station must use the facilities of other stations to perform these actions, it will coordinate the contacts which must be made to other DCS stations and command levels.

The procedure for selecting circuits to pre-empt for altrouted circuits is stated as follows:

(1) Proceed down this list in order to pre-empt circuits:

(a) Spare circuits.

(b) Non-activiated on-call circuits.

(c) Circuits with RPO or none at all.

(d) Circuits with RP 4C and proceeding up the letter and number RP categories.

(2) Do not altroute one AUTOVON trunk or access line over another when they connect to the same points.

(3) Reroute teletypewriter tone packages before voice circuits when both have the same priority.

The guidelines to be observed when establishing an altroute are given in DCA circular 310-65-2 (Ref. 6) and are listed here as follows:

(1) Use the shortest mileage possible.

(2) Use the minimum number of trunks in tandem.

(3) Make the distance between point of entry to a group or supergroup to point of exit as long as possible.

Finally, DCA circular 370-70-5 (Europe) seems to summarize the other documents with regard to altrouting. It calls for RP #1 circuits to have pre-planned altroutes on backbone links or as directed by CINCEUR. In the event that restoral via altrouting is not planned on a failed circuit, follow directives in 310-70-1.

2.1.2 Comments on a DCS Site Visit

In spring of 1978, researchers for the MItre Corporation visited European sites of the DCS for the purpose of studying operations. Their goal was to study ways of implementing system controls to improve the DCS operations. Some of their observations (reference 7) are related to altrouting and are summarized here. They provide real justification for embarking upon development of automated altrouting in the DCS.

The primary problem with circuit allocation noted by the Mitre report is a data base that is sometimes found to be faulty and the lack of computer-aided tools for circuit routing. They explain that the current circuit design tool (CACFAS) does not help circuit engineering if trunks or circuits do not already

exist over the route being examined. The engineering then must be done manually with multiplexer maps and trunk and circuit lists. They recommend a new computer-aided design tool to search out circuit routes. This tool might be applied to the network to re-engineer all circuits due to the poor routing that has evolved over the years with the current engineering system.

Their comments on restoral actions following link failures points out another problem area. There are only 12 altrouting plans now recorded for all of Europe and these are out of date. When altrouting actions must be taken, therefore, considerable coordination activity must occur between TCF's and FCO's. A lack of effective inter-service coordination seems to be making this altrouting action difficult under current operations.

2.1.3 Summary of the Current ALtrouting Procedures

The DCA documentation studied seems to provide merely a basic framework in which to undertake altrouting actions. There is some guidance as to what a desirable circuit route should look like and some rules for deciding pre-emptability of circuits for altrouting purposes. There does not seem to be firm guidelines on coordination among stations working on altrouting. This seems to be a problem area as reference 7 points out.

The action during altrouting is directed upward from the lowest level of the DCS command structure. This method seems to lead to the inter-service coordination problem noted in reference 7. It also concentrates the bulk of the altrouting responsibility on the levels that have the least access to current global data about network availability.

The net result appears to be that altrouting currently is a function of the quality and cooperativeness of the station controllers. These controllers must deal with the huge circuit and trunk data base without any automated aid. They must resolve conflicts over planned actions over inter-service barriers. In short, they must have a well developed personal "feel" for the network and their station comrades.

2.2 ALTROUTING ON THE DCS IN THE 1980'S

The DCS of the 1980's is beginning to shape up into a network where automated algorithms can be implemented to carry out the altrouting directives that are required but are not possible now. Improved data base access and the quick status reporting of a new ATEC system in those years should make the network seen by the controllers more realistic. This would mean that altrouting planning could be done more readily and an automated algorithm would be just as knowledgeable about the network as the station staff. The digital nature of the network in that time period also makes it easier to locate compatible channels, allowing more connectivity possibilities for altrouting. Finally, automated

patching at stations would allow quick response to altroutes developed by the automated algorithms. Thus, is it beneficial to have an algorithm that can turn up a large number of altroutes before equipment repair occurs.

2.3 INTRODUCTION TO THE ALTROUTING ALGORITHMS

The information of section 2.0 to 2.2 and the discussion there points to the need for an automated altroute search algorithm and the feasibility of such an algorithm working. This section takes this premise as a starting point. The background found in section 2.1 provides the altrouting guidelines of DCA procedure to work within. Comments from reference 7 provide points to improve upon in the automated algorithm. The resources of ATEC and the data base designed in reference 2 provide the tools to develop a search algorithm.

The first topic of this section is a discussion of the altroute search technique used. This is a key issue in the development because it is the tool that keeps the search from becoming a brute-force exercise in data base searching.

The next major area of this section is a discussion of the concepts used in the algorithm. By this we mean, the way in which the DCA policy guidelines are implemented and the additional rules developed where no policy currently exists. In addition, discussion is made of how the algorithm can incorporate the knowledge of the tech controller in its approach to the search and provide flexibility to changes in policy of DCA operations.

Finally, the algorithms themselves are presented. Condensed as well as detailed flow diagrams are presented. A discussion of the operations is made and the key data base needs are pointed out.

The last part of this section is the analysis of the benefits of the use of the algorithm. The result obtained is an availability of service for a circuit under conditions of possible equipment failure and restoral via altrouting or repair. Assumptions are made as to the composition of the circuit groups to be rerouted in the analysis so as to provide an average benefit of the algorithms.

2.4 SEARCH TECHNIQUE

The establishment of an altroute is the problem of finding a path through the network between appropriate break-out points of the circuit or trunk. In addition, we will require that the path found is the lowest "cost" path that exists between the two points of interest. This is the goal of the search algorithm to be developed. The approach used to arrive at the lowest cost path involves node labeling procedures that are common to almost all search techniques in graph theory. The basic idea is to

branch outward from each node in the network that has been accessed already and determine the cost to the next accessible node. That node is labelled with the cost of that transition plus the cost to get to the node just left. Pointers to the node previously labelled are used to establish the path linking of the two nodes. The optimal cost is achieved by linking to the current node that accesses a new node to be labelled if the cost that would be assigned by that node is lower than the cost already assigned (as a result of access from some other node). Labelling from both ends of the path is one variation that can speed the process.

An algorithm based upon the idea described above does not use any intelligent information about the network it searches. It is totally unbiased in its node labelling and will probably label nearly every node in the network. We seek an algorithm that can utilize some knowledge of the network which will reduce the number of nodes that need to be examined in the search process. Heuristic search techniques (Ref. 8) do just this and are used often when the number of possible nodes to examine is large and the number actually examined must be reduced.

The heuristic search routine differs from the unbiased routine described earlier in that it not only finds the cost of the path to the node it has just added to the path, but estimates the remaining cost to complete the path to the other end from that node. In this way, the cost at each node reflects the total path cost of a path through that node all the way to the other end. It also requires that the program have some good estimating routine for seeing into the future of the path at each node. The ability to provide this heuristic forward information differentiates this type of routine from the brute force search often used.

The real importance of the heuristic cost estimate of the remainder of the path must be emphasized. A routine that only computes the cost of the path to the last node added will result in many possible paths being examined before one finally reaches the path end. The reason for this is that the path pieces which are short and have not yet been expanded will soon have a cost less than the piece which is now longest. It also means that, based upon cost, the shortest pieces must now be examined rather than continuing on the longest path piece. The net result is that all possible path pieces will be approximately the same length at all times. This means that nearly all nodes will be examined in the search. The use of a heuristic cost estimate for the remainder of the path makes each node's cost approximate the total cost of the path. This means that the longest path expanded will not necessarily have the largest cost. Thus, if it is lowest in total cost, it will continue to expand and prevent examining excess nodes. Of course, if at some point it does cease to be least costly, other path pieces will be examined ahead of it for further expansion. The bottom line is that a path piece that appears promising will not be hindered in being expanded just

because it is ahead of the other possibilities in length. Finally, it can be shown that if the heuristic cost estimate never exceeds the true cost of the unknown path segment, then the lowest cost path will always be found by such a search.

With these thougths in mind, the following algorithm is presented as the heuristic search algorithm which makes up the heart of the altroute searching algorithm.

(1) Put the start node(s) on a list called OPEN and compute the cost for that node.

(2) If OPEN list is empty, exit with failure; otherwise, continue.

(3) Remove from list OPEN the node with the smallest cost and put it into a list called CLOSED. Call this node n.

(4) If n is a goal node, exit with success and find the pandh by linking the parent stations of n. If n is not a goal, continue.

(5) Expand n into all nodes it has access to. (If there are no such stations, go to (2).) For each node accessible from n, compute the cost.

(6) For successors of n that are not in either the OPEN or CLOSED lists, enter into the OPEN list along with its cost and a note that n is its parent station.

(7) For those successors of n that are on either OPEN or CLOSED lists, compare the cost already listed with the one found from n. If the cost is less, go to 2. Otherwise, enter n as this node's parent station and enter the cost found from n. Put that node in list OPEN if it is not already there.

Finally to relate this abstract graph search algorithm to the question of altroute searching, the following analogies are needed. The nodes of the graph are stations of the DCS network. The expansion from a node involves finding the next trunk or circuit level access stations that can be reached by the reroute group via pre-emption of less important circuits or trunks. The cost of accessing this station is the new pre-emptions required, distance of the link, patches required and other items of knowledge about the link to the new station. The goal is a station where circuit or trunk access is available and where connection to the final circuit or trunk destination is still intact. With these analogies, it remains to decide how to implement the seven steps for the DCS network in particular.

2.5 CONCEPTS OF THE ALGORITHMS

2.5.1 Assumptions

Before starting a detailed discussion of the key algorithm concepts, a list of the assumptions made is in order. These are in addition to the network assumptions made earlier in the report. The added assumptions are as follows:

(1) We will assume that patching is possible at the circuit, group and sub-VF circuit levels. This assumption will be made for all stations in the network.

(2) The ATEC system monitoring function is assumed to be fully deployed. This means that the data base used in the altroute searching will be up-to-date. It also means that inputs from faults that have just occurred can be used to trigger the algorithm into action. The fault isolation feature of ATEC should also provide stations bounding the failures being reported.

(3) We will assume that the data base used in the search is the one designed in reference 2 and updated as shown in Appendix A. All transmission facilities used in the altrouting must have their data in the form of this data base, otherwise they cannot be used by the algorithm developed here.

(4) A system of Restoral Priorities (RP's) must be in effect to guide the algorithm in its pre-emption activities. This system may need to be an expansion of the current system so that levels of circuit importance carried by function, agency or specific DCA directives (Ref. 3) are shown directly in the circuit's RP classification and not derived by any other unwritten criteria.

2.5.2 The Altrouting Hierarchy

The altrouting hierarchy describes the various levels that exist in the algorithm for entering the search and for reconfiguring the search if a higher level of search failed. The two main hierarchies to be discussed are the multiplex hierarchies and the fault isolation structuring.

The entry to the algorithm developed here is made at the highest level of altrouting coinciding with the reported equipment failure. That is, we attempt to altroute a trunk if a group carrying the trunk fails rather than altrouting the circuits the trunk carries individually. Likewise, circuits carrying sub-VF circuits are altrouted before their individual sub-VF components.

This results in fewer patches and data base updates for an altroute when it is possible to altroute at this highest level.

When failure to altroute occurs at the group level, the trunk is broken into its circuits and circuit-level altrouting is attempted. This is true for VF trunk and sub-VF level circuits. Those circuits that were on the trunk may have different end points, but they do appear at circuit level at the trunk ends. Thus, if the group is altrouted from one trunk end to the other, the altrouting search can handle them all together rather than repeating the same search (over possible the same altroute) for each one individually. This concept is used in the transition from trunk to circuit level search hierarchies in the algorithm. The algorithm will altroute the most important members of such a reroute group and drop off group members which cannot be carried along the route of the most important circuit in the group. Those group members discarded from the altroute will be used along with the partial altroute (found before they were discarded) to re-enter the altroute search.

The fault isolation structure has already been mentioned when we decided that circuits on a group should attempt to be altrouted between the trunk's end stations. The common fault of the group was isolated to be bounded by the trunk ends. This was done not only because it was the only known common station of circuit level access of the group, but because it was the first station bounding the known failure where circuit level access points to the remaining intact parts of the normal routes is available. Similarly, individual circuits or trunks entered to the altroute search will be examined in order to isolate the faults on their normal transmission facilities and then define the remaining route segments that are intact. These fault boundry stations can then be used to start and end the altroute. When failure to link the stations bounding the failure occurs, the algorithm begins the search from a station one step closer to the circuit's or trunk's end station. This provides a new connectivity for the station beginning the search.

The fault isolation structure is also evident when a failed link forms a dead end path for the failed segments of the circuit or trunk. The goal station definition routine searches out such dead end paths and moves the starting and goal stations of that path so as to prevent "backhauling" from a dead end path.

Further discussion of the multiplex hierarchy levels is made in section 2.6.1. The details of the isolation fault structuring is presented in section 2.6.5.

2.5.3 Selecting Pre-Emptable Altroutes

The key to finding an altroute is the ability to find circuits that are less important than the circuits to be altrouted and use their transmission facilities by pre-empting them on those

facilities. It is true that spare facilities should be used before this tactic is used, however, the spare facilities will probably not accommodate all the altrouting needs in the event of link or group equipment failures.

In order to carry out pre-emption in a manner consistent with DCA directives, an enhanced version of the circuit RP system will be used. As assumed, this system must incorporate all known relative importance items of circuits. The algorithm cannot be called on to make individual choices in pre-emptions without this system. Conversely, it is through this RP system that operations personnel can manipulate the algorithm by subdividing RP's according to the importance they see in the circuits.

The pre-emption of individual circuits by other circuits will be done exactly as recommended - the higher RP always can pre-empt the lower RP circuit. Trunk or group altrouting will require that the RP's on the altroute group or trunk must exceed all RP's on the group or trunk must exceed all RP's on the group or trunk to be pre-empted. This is consistent with the individual circuit rules. Spare circuits or groups should always be given RP classes which are lower than all other in service circuits or groups so that altrouting will proceed to these first.

The rule on trunk or group pre-emption stated above will probably restrict altrouting at this level to spares. It should be noted that being less stringent about the rule and allowing pre-emption when the great maj rity of the altroute group RP's exceeds the possible pre-emption group's RP's may speed the altrouting of a large number of circuits at the expenses of a few high RP circuits. These high RP circuits may be altrouted again to restore service right after the group is altrouted. This policy promotes a volume approach to the altrouting problem rather than the strict highest priority first approach used today.

2.5.4 The 'Cost' of an Altroute

The key to making the search algorithm efficient is to provide it with the same inputs a clever tech controller would use in establishing an altroute. The vehicle for this data is the "costs" assigned to each segment along a possible altroute and the heuristic estimates made for the path beyond the current partial altroute. With these tools, the algorithm can compute the cost of the partial altroute it has and see the mux maps and know the network characteristics as the tech controller would to plot the next segment of the altroute. If this principle is followed, the algorithm will do the tech controller's job, but at computer speed and accuracy.

Costs for the partially established segment of the altroute are easiest to obtain because we know the route at that instant. Costs that are probably similar to what the tech controller actually subconsciously uses are:

(1) The mileage of the route.

(2) The PP's of the circuits pre-empted to use the route.

(3) The number of patches made to connect the partial altroute.

(4) The type and location of the transmission facility used.

The heuristic estimates must use these same type of costs, but estimate them for the segment of the altroute which is not known at this instant. This is considerably more difficult. Also, remember that if these estimates do not exceed the true cost of the rest of the route, then we are guaranteed of the lowest cost altroute. However, if the estimates do exceed the true cost, a faster search in some cases is possible. We adopt the more conservative approach and try to keep the estimates below real costs.

The following discussion centers on the cost to be used and how they are calculated for the known altroute segment and estimated for the unknown altroute segment.

(1) The Route Distance.
The simplest way to find the route distance without increasing the data base is to use the Connectivity Path File (CNF) described in reference 2. We propose to modify the file to include all stations on each "path" listed and the mileage between those stations. Thus, to calculate the mileage of the path found up to the current station, simply add the mileages from the CNF for the paths traversed.

The use of the CNF will work fine unless a station happens to not be on one of the paths. Then, we propose having a new Station Coordinate File (SCF) with coordinates of each station. A station off of the paths may have a distance computed for it by adding the distances along the route for which there is path mileages in the CNF to the Euclidean distance from it to the last station along the route that was on a path.

The heuristic cost of distance to the end of the altroute over unknown links can now be found with the use of the CNF and SCF. The Euclidean distance of the ends of the paths that the goal station and present station are on can be computed from the SCF of the path ends. The distance to those ends for both paths can be added to this distance from the CNF. The result should be an excellent lower bound on the true

length of the remaining path segments. For stations not on a path, simply use the Euclidean distances found from the SCF.

This distance cost is likely to be the most important cost of the total because it is the way that the algorithm sees the mux map and determines the direction to be taking. It is probably also the first and foremost criteria on a possible altroute that a tech controller would use.

(2) RP's of the Pre-Empted Circuits.
Little need be said about this cost for the portion of the route that is known - simply add the RP's of all circuits pre-empted along the route to this station. Use sum of the RP's for trunks carrying many circuits.

There does not appear to be any way of finding this cost for the part of the route not yet determined. For this reason, care must be taken in weigthing this cost with other costs that have heuristic estimates for the unknown segment of the altroute. As was mentioned earlier, the effect of an unestimated cost is to make many search paths of the same length rather than concentrating on the most probable path. Of course, the cost is still important as part of a measure of a good altroute and the decision of how to balance these conflicting results of this cost will need to be made with actual examples of the algorithm's results.

(3) Number of Patches Required.
The total number of circuits or trunks pre-empted to the current station is the value of this cost. This will be the number of times a patch was made to form the partial route.

Again, this cost does not lend itself to estimation for the unknown segment of the altroute. The same warning for weighting this cost with the others applies.

(4) Transmission Facility Costs.
The mainline DCS transmission medium of the 1980's is line-of-sight microwave, although cable, tropo radio and satellite links are present. The desirability of using each type of facility in an altroute should be considered. In addition, the reliability of individual pieces of equipment should also be considered in the altroute plan. Both of these items are certainly considered by the tech controller when the altroute is planned. For these reasons, a cost will be assigned to each type of transmission facility which should take

into account the relative desirability of that facility in its own class of equipment and over all types of transmission facilities. Note that this cost is another way that control over the algorithm's search technique is provided to operations personnel.

The obvious place to store such information is again the Connectivity Path File. The cost for the portion of the route that is known is again found by simply adding the station-to-station transmission costs along the path. Segments without path listing can have some average transmission cost per mile applied. The estimate of the unknown segment transmission cost can be done similar to the unknown segment mileages. Use the CNF costs for points to the closest ends of the paths that the current station and the goal station are on. Use some average cost per mile times the Euclidean distance between these closest end stations. If no path is attached to the stations, simply use some standard factor times the Euclidean distance.

2.5.5 Altrouting in a Network with AUtomated Patching

The 1980's DCS may well have equipment at each station capable of automated patching. By this we mean that the routing of circuits is carried out in PCM form and that any circuit entering the station will be able to be routed to any other link or group to leave the station. The purpose of this discussion is to study the impact of such flexibility upon the search algorithm. The main effect of the algorithm will be that there will be a guaranteed direct route from any station to any other station. This route will also not have any "backhauling" or loops. Perhaps this is not obvious, so let us consider the network as it exists now. Today's network defines trunks which carry a number of circuits and which provide break-out for those circuits only at the trunk ends (where the channel banks are). The route of most trunks is often over several stations. This means that the circuits can not be broken out at any point we wish. We must go to the trunk end and then return to some station along the trunk's route via another trunk if we wish to terminate and patch circuits at some intermediate station. The result is that "backhauling" is often needed and in some highly connected regions of the network, loops made need to be formed just to access the right trunk to reach a particular station. The current DCS data base for Europe shows numerous examples of this.

The use of the automated patching device described earlier would allow circuits to be broken out at any station and eliminate the problems of "backhauling" and looping and further guarantee a direct route to any station in the network.

The effect of the automated patching on the algorithm's search is great. In the current trunked network, the algorithm must examine

every trunk from the station over a link to make sure that all possible new stations can be reached. The reason is that we are not guaranteed that the closest accessible station will get the route to the goal station. Thus, all possible ways of leaving the current station must be considered. The automated patch network, however, can simply find one route to the next station and end its search at that station over a link. We are guaranteed that we can find a route to the goal from that one next station. The result is that fewer file accesses are required by the algorithm t search the current station for path expansion. This is crucial in the time required to run the algorithm.

The routes which the altroute takes are different in the two networks. The trunked network will no doubt have multi-station jumps in the altroute which will sometimes provide a quick route and other times require "backhauling". The automated patch network, which is link oriented, will move directly station by station to a solution. It will sometimes require more station patches and sometimes fewer due to its direct approach of the goal station. In the end analysis, it seems that the guarantee of a direct altroute is a worthy benefit to the current somewhat circuitous routing seen in DCS circuits today.

2.5.6 Movable Transmission Resources - Satellites

The majority of the links of the DCS are fixed in both end stations and channel capacity. An exception to this is the link provided by a satellite. The end stations can be selected from the group of stations at the earth terminals. Although the trunks or groups to the earth terminal from the DCS is a fixed capacity resource, the channels to a particular earth station over the satellite links is variable. This means that if circuits at a receiving earth station from some other earth station can be pre-empted by the altroute circuit group or trunk then those channels can be moved to the current search earth station to increase the capacity to the receiving earth station. In other words, the circuits pre-emptable over a satellite link are not only those currently present at the station, but those reaching the receiving earth station from any other earth terminal.

Since the satellite network of the DSCS in the 1980's should be able to provide dynamic channel allocation among earth terminals, this resource of variable channel capacity will be exploited in the altrouting algorithm. To prevent over use of this option, only when no pre-emptable circuits over the normally assigned channels is encountered will the algorithm request moving additional channels to the current search earth station.

2.5.7 Returning to Normal Routing

Once the failed equipment along the normally assigned route of a circuit or trunk is repaired, the circuit or trunk can return to

that route. ATEC will provide the triggering for this process as it deletes the last fault file assigned to a trunk or circuit. The algorithm will then check the normal route to see if return is possible. This check involves determining whether any portion of the normal route was pre-empted by another circuit while the normal circuit was altrouted. If a pre-empting circuit is still found on the route, return is not allowed. The reason is that the pre-empting circuit will always be of higher RP and return to the route would violate the directive of only pre-empting lower RP circuits.

One possible consequence of this rule would be the return after a large portion of the DCS was failed. Those circuits altrouted earlier due to the first of the failures would be ready for return to normal before the latter circuits that were altrouted due to the last failures. In the event of large DCS failures, the normal routes of the first altrouted circuits would likely be used as parts of altroutes for circuits altrouted latter on. Thus, the circuits first to require return would need to wait until nearly all of the failed network equipment was repaired. This would result in a "domino" effect in the restoral of normal routes in the network. This scenario is unavoidable and a direct result of the RP pre-emption rule.

2.6 THE ALGORITHMS

2.6.1 The Main Calling Routine

2.6.1.1 Discussion of the Routine --The main calling routine is the part of the algorithm which is entered when an ATEC fault report occurs. It is in this algorithm where the input from fault reporting is stored and appropriate altroute searches are begun. This routine is also where the search routines return with their results and decisions are made at that point as to what action is to be taken. Section 2.6.1.3 provides detailed flow charting of this routine.

The entry points for the routine are the three levels of equipment failure discussed earlier. The routine stores the failed service that ATEC reports and the stations along the route of that service bounding failed segments. These segments are assumed to be failed in both directions by the routine - altrouting only one direction is not desirable due to the possibility of dropping the other direction during repair or test. The order in which the calling routine proceeds on its stockpile of service to altroute is according to the maximum good that an altroute will do. Thus, trunks are altrouted before circuits and circuits before sub-vf circuits. In the class of trunks, the trunks are ordered according to the sum of the RP's of the circuits they carry. The top sum of RP's trunk is altrouted first, and the others in order of decreasing sum of RP's. The same technique is used for circuits.

The routine begins by passing the stations bounding the failed segments of the service to the search routine. The search routine defines two segments of the route that are intact and as close as possible to the failed segment. Stations OS and FS define the longest segment and ITS and TS define the shorter segment. The search preceeds from FS to any station between ITS and TS with the proper patching access. More about this definition is made in section 2.6.5. If the initial choice of these starting and goal stations fails to yield an altroute, the search routine returns to this calling routine with that information. The calling routine then will shorten the longer intact segment by one station. This gives the routine a new starting point (FS) which may provide better connectivity for an altroute.

If the search from the circuit or trunk ends yields no altroute, then failure to altroute at that level is declared and attempts are made to break the trunk or circuit into lower level components for altrouting.

Failure to altroute a trunk on the trunk level does not mean that an altroute for most of the circuits being carried cannot be found. It just means that on that group multiplex level, no pre-emptable or spare groups were found to create an altroute. Perhaps by dropping down to the circuit (or channel multiplex) level, an altroute can be constructed. This is exactly what the routine does at this point. Circuits very often traverse a route longer than one trunk, thus it advantageous to allow the search for a circuit altroute to look at many possible starting and goal stations along the route as possible. In addition, several parallel trunks can be called upon to create the altroute by yielding their lowest RP circuits for pre-emption. This creates a larger chance of finding an altroute than when the trunk must be kept together. The trunk is decomposed into circuits and the circuits are ordered by their RP's. The list of these circuits is now in the same list that individual circuits would be entered if they had failed alone. The RP ordering determines which circuit will be worked on first by the search routines.

When a circuit at the top of the circuit altroute list is selected to altroute, a search is made to see if any other circuits in the list share its same failure boundary stations (i.e. FS and ITS stations). If so, these stations are added to the top circuit to create a group of circuits which have at least two points in common (FS and ITS) where an altroute is needed. These circuits are altrouted together when the search routine is called. A minimum RP (RPO) is applied to the group so that the operator can set the lowest RP circuit of each group which should be altrouted before the group is disbanded. More will be said about this group altrouting concept in section 2.6.3, but suffice it to say that this is a service the calling routine does for the

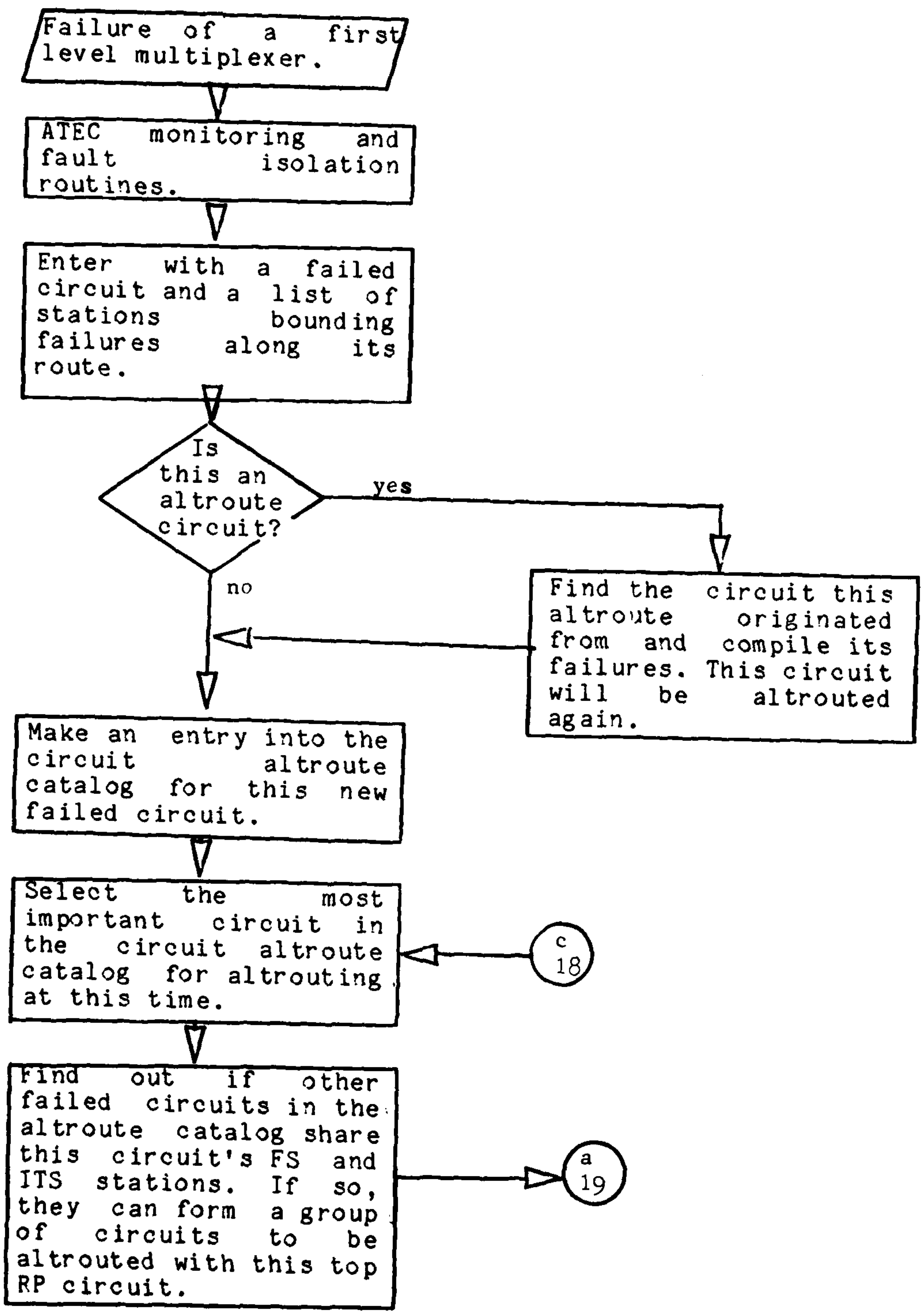
search routine. The same, of course, is done to circuits carrying sub-vf level circuits.

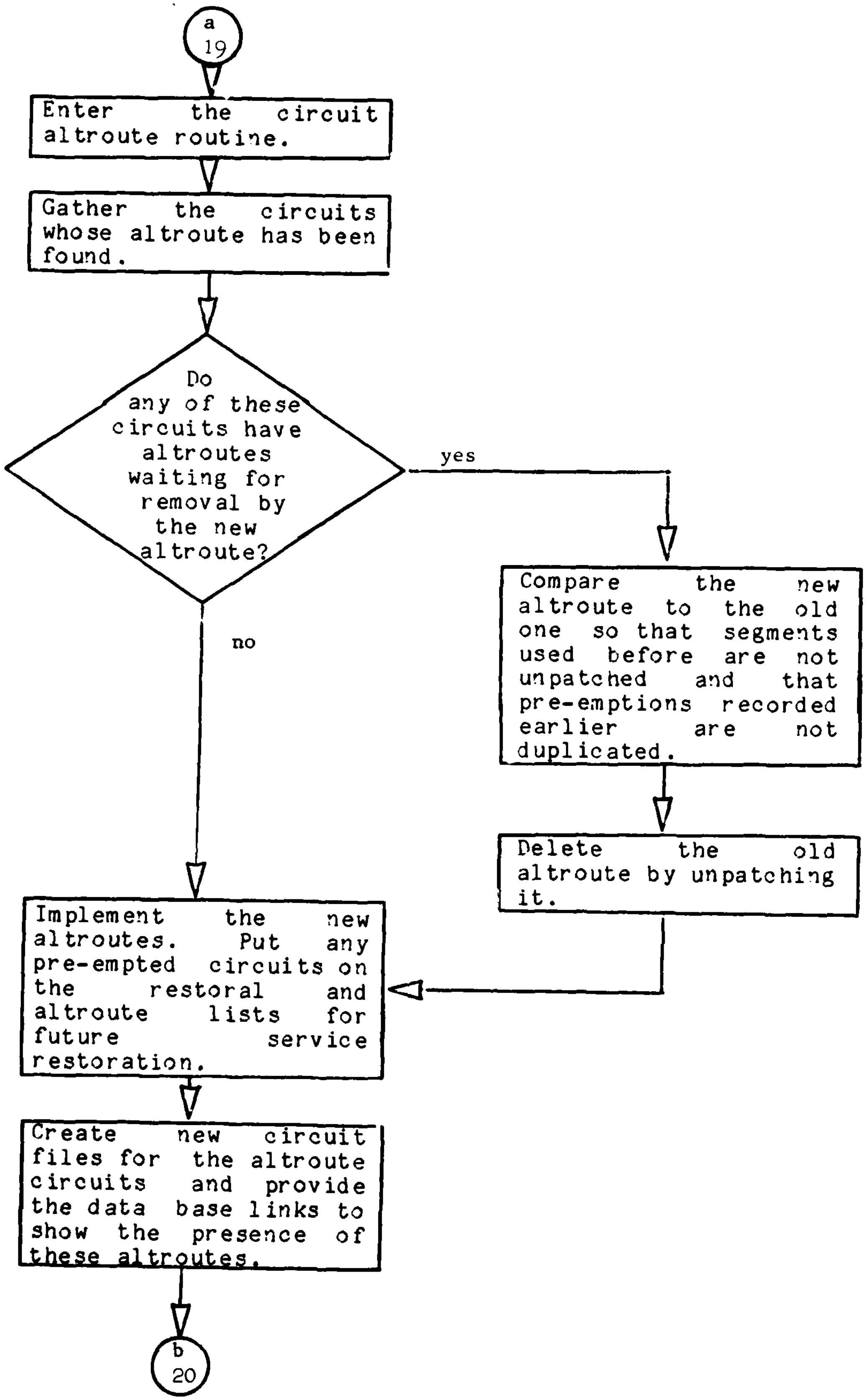
Finally, a successful altroute search will provide the list of patches to make and circuits or trunks pre-empted which should now be considered for altrouting themselves. The calling routine handles this function and provides a file ALTP with the altroute data.

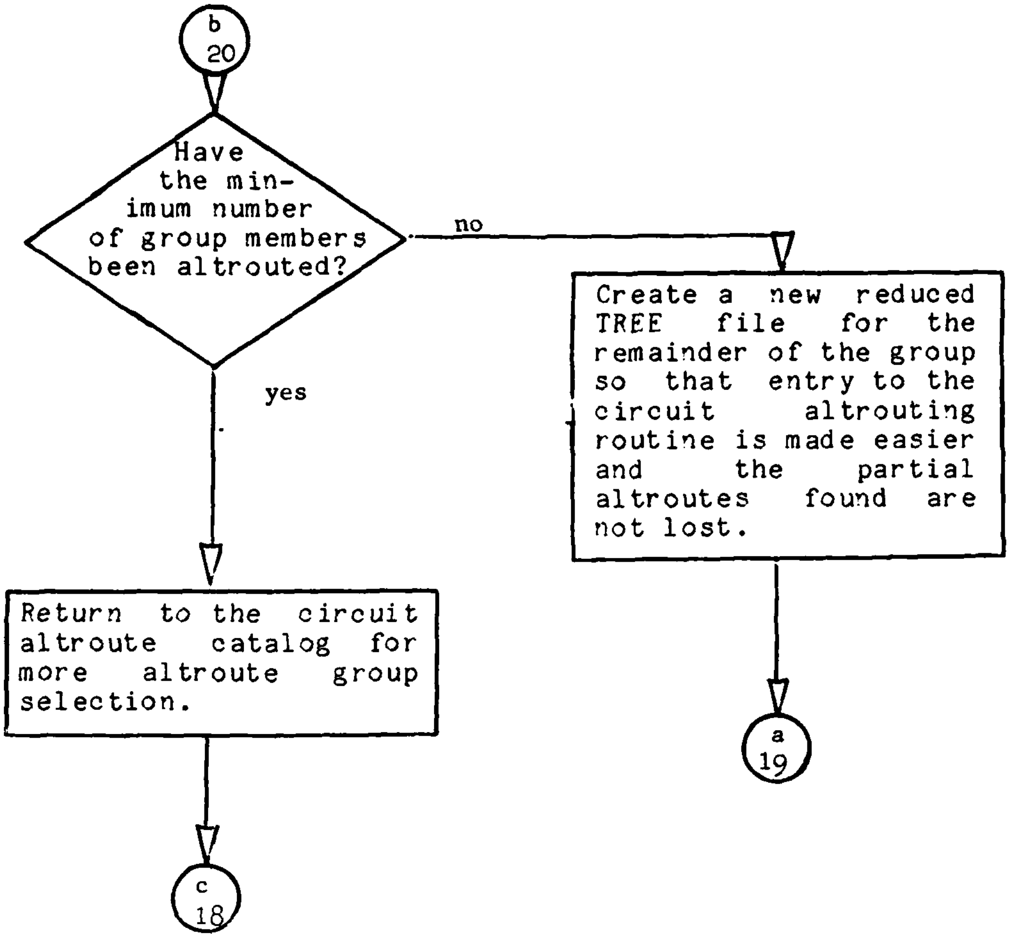
In some cases, an altroute may fail before the normal route of a trunk or circuit can be restored. There are two ways of handling this event. One way is to treat the altroute like any other circuit or trunk and altroute it. The other approach (and the one adopted here) is to go back to the service the altroute evolved from and altroute from that point. The reason the second method is used is that the network being considered is not highly connected. Thus, an altroute of an altroute may not just by-pass the failed segment, but actually find a very different route which leaves portions of the first altroute in place but unused. This scenario would be serious if the failure in this case is a link and large number of unuseable circuits or trunks are now left over from the first altroute. The loss of transmission facilities would be large and be a serious impact on further service altrouting. With the removal of the first altroute, the transmission facilities used on the first altroute are freed for use by the pre-empted circuits or trunks. It is likely in some cases that the new altroute would use some of the segments of the old altroute. Thus, unpatching of the old altroute is not attempted until the new altroute can be compared to it. Only when common segments are identified, can the old altroute be unpatched.

To aid in understanding the main calling routine's logical flow, a simplified flow chart is presented. This flow chart outlines the key areas of the routine for the circuit entry section. The trunk and sub-vf sections of the main calling routine are so similar to the circuit section, that only one section need be simplified to understand the flow of the other sections.

Simplified Flow Chart for the Circuit Entry Section of the Main Calling Routine




a
19
Enter the circuit altroute routine.
Gather the circuits whose altroute has been found.
Do any of these circuits have altroutes waiting for removal by the new altroute?
yes
no
Compare the new altroute to the old one so that segments used before are not unpatched and that pre-emptions recorded earlier are not duplicated.
Delete the old altroute by unpatching it.
Implement the new altroutes. Put any pre-empted circuits on the restoral and altroute lists for future service restoration.
Create new circuit files for the altroute circuits and provide the data base links to show the presence of these altroutes.
b
20


b 20
Have the min-imum number of group members been altrouted?
no
Create a new reduced TREE file for the remainder of the group so that entry to the circuit altrouting routine is made easier and the partial altroutes found are not lost.
a 19
yes
Return to the circuit altroute catalog for more altroute group selection.
c 18

2.6.1.2 <u>The Required Data Base</u>--This routine requires trunk and circuit files as detailed in Appendix A. The only other file that is not self-explanatory is the listing of trunks and circuits requiring altrouting. Figure 2-1 details the items needed in this file.

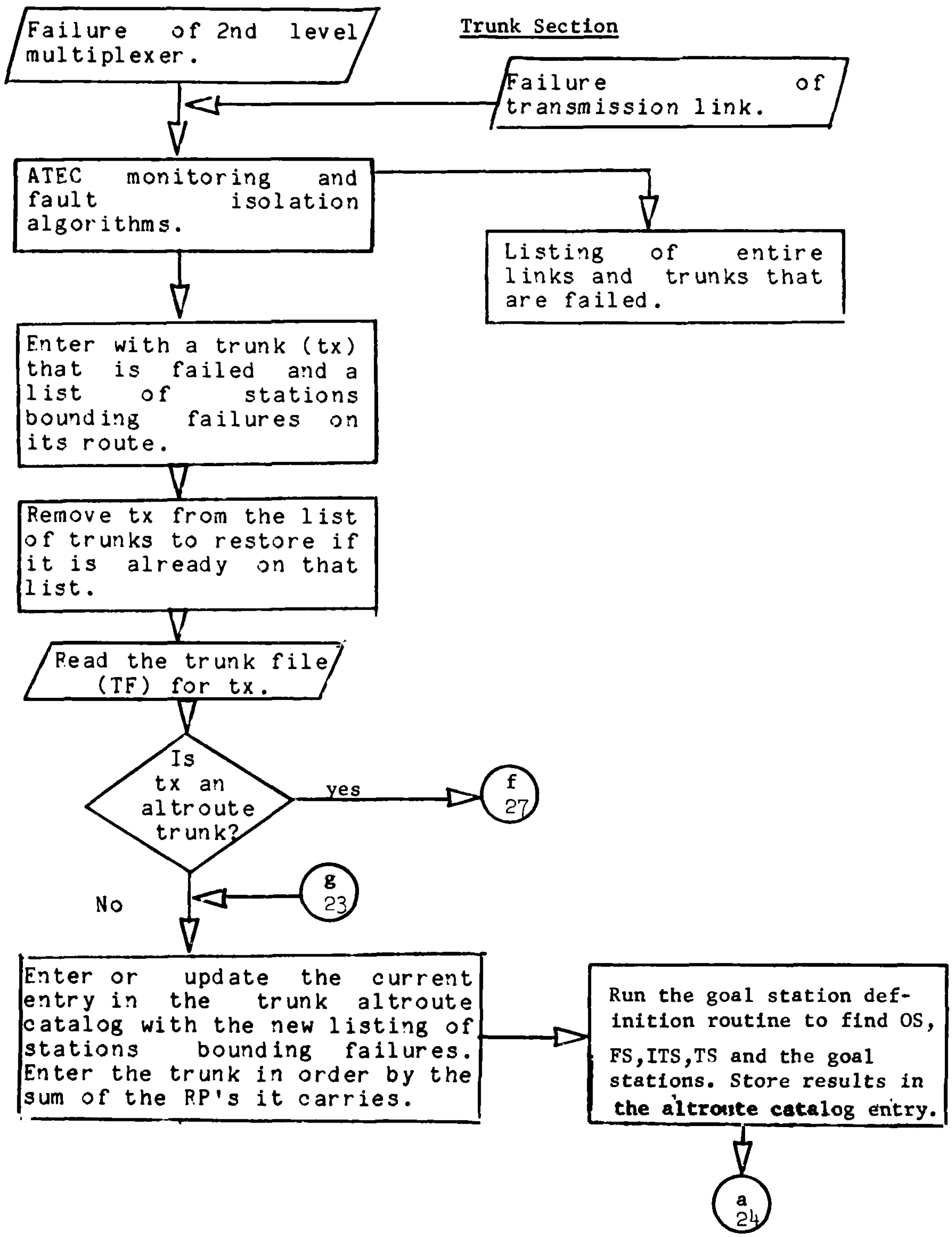
Trunk or Circuit Needing Altrouting	. . .	Stations along the route bounding failures (9 max.)	. . .	Terminal end of biggest segment (OS)	End of the largest intact that is closest to the failures (FS)
(i) 8 bytes	(2)	18 bytes		(3) 3 bytes	(4) 3 bytes

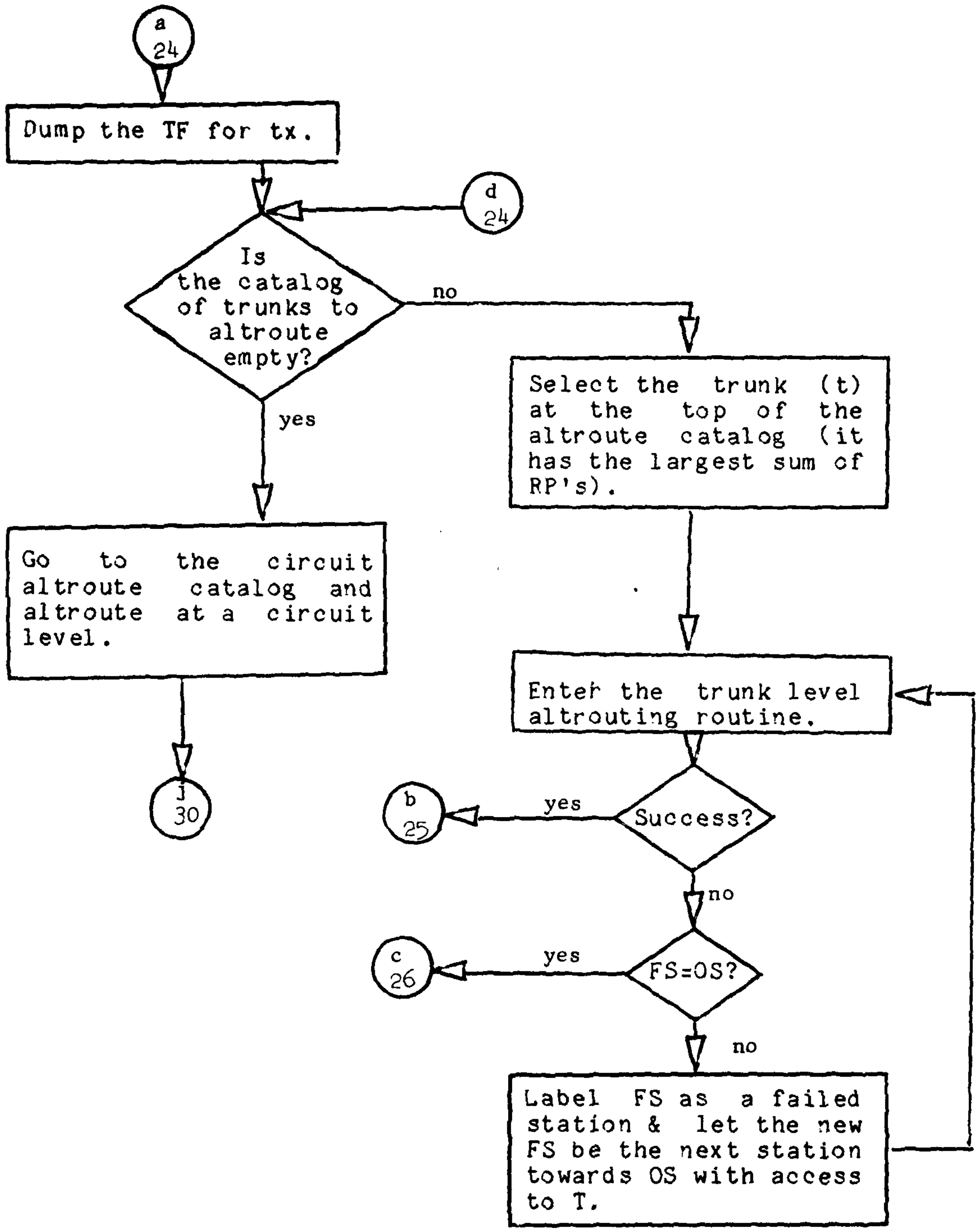
Figure 2-1

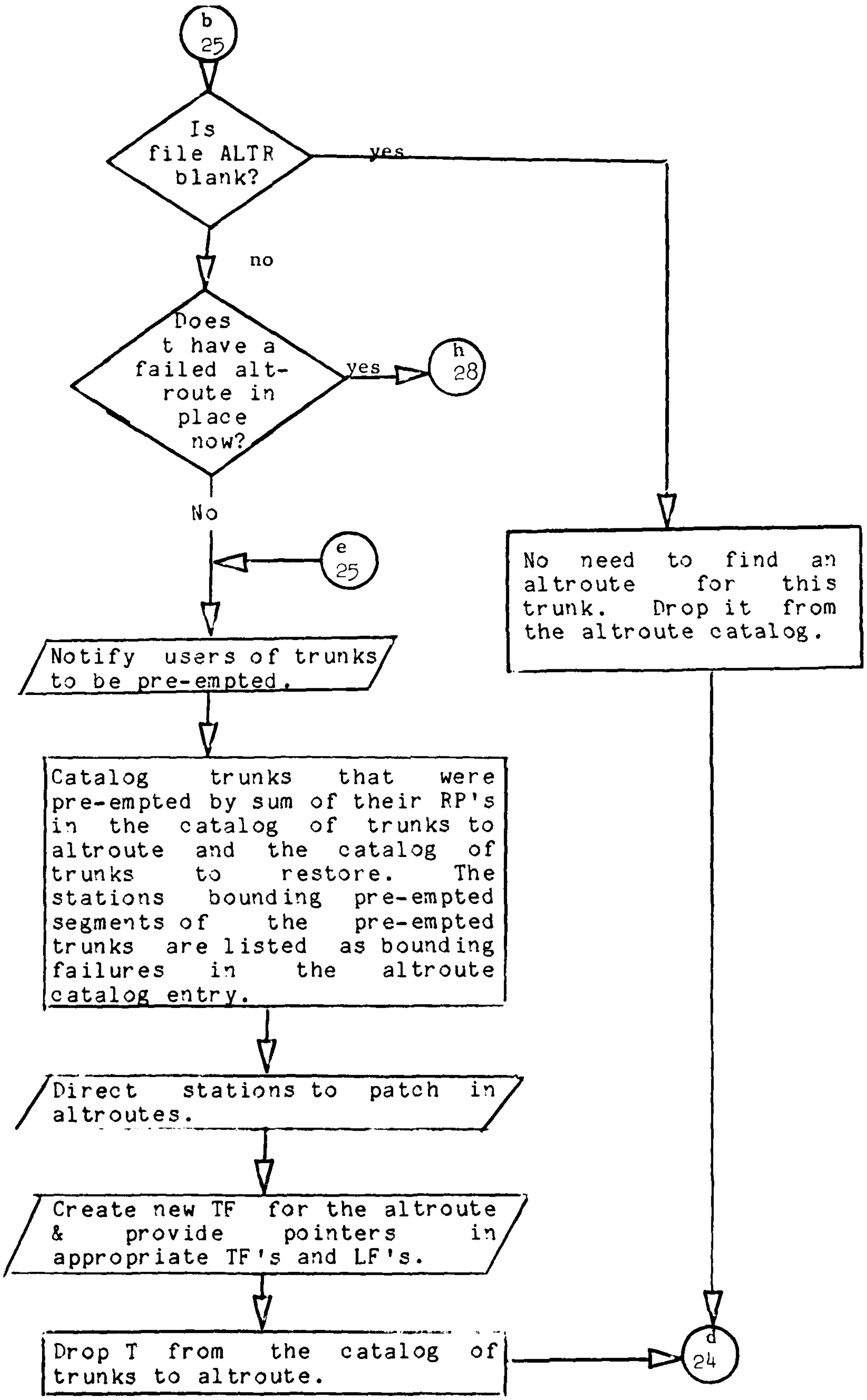
The Structure of the trunk/circuit altroute catalog

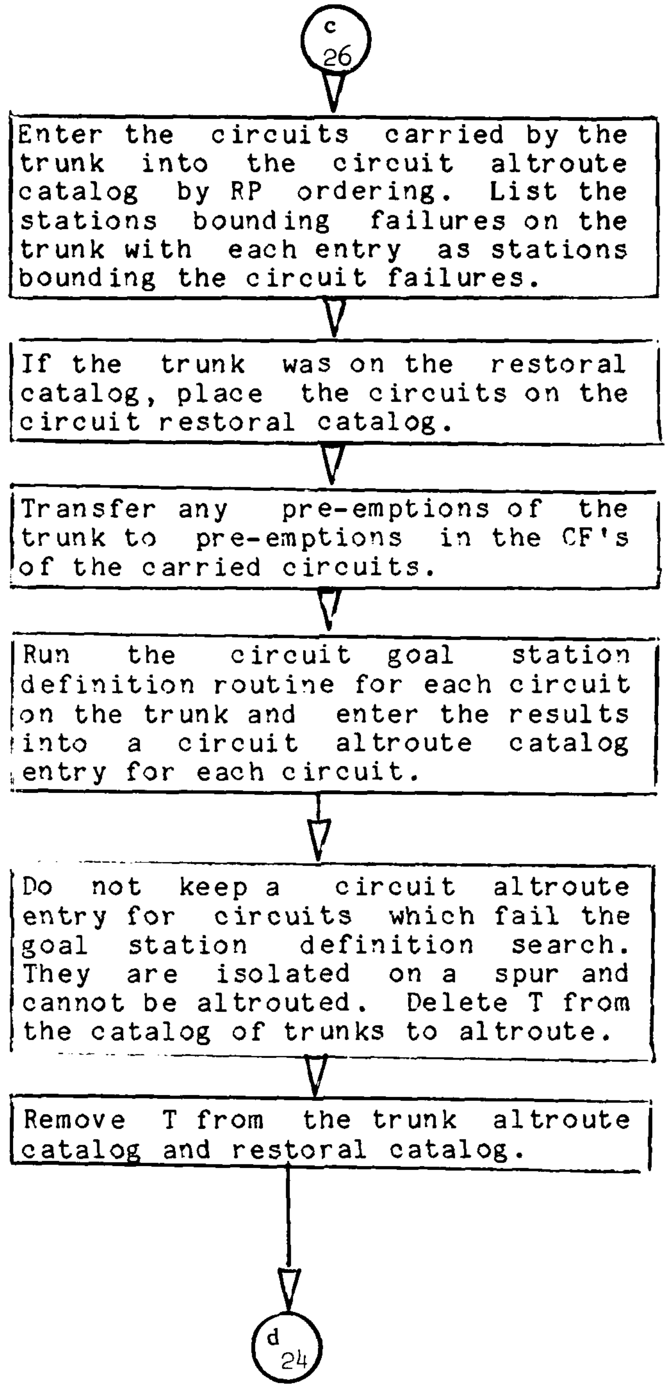
End of the shortest intact segment that is closest the failures (ITS)	Terminal end of the shortest intact segment (TS)	. . .	Goal stations-station between ITS & TS with access to the service (6 max.)	. . .	Port type used for this circuit.
(5) 3 bytes	(6) 3 bytes		(7) 18 bytes		(8) 1 byte

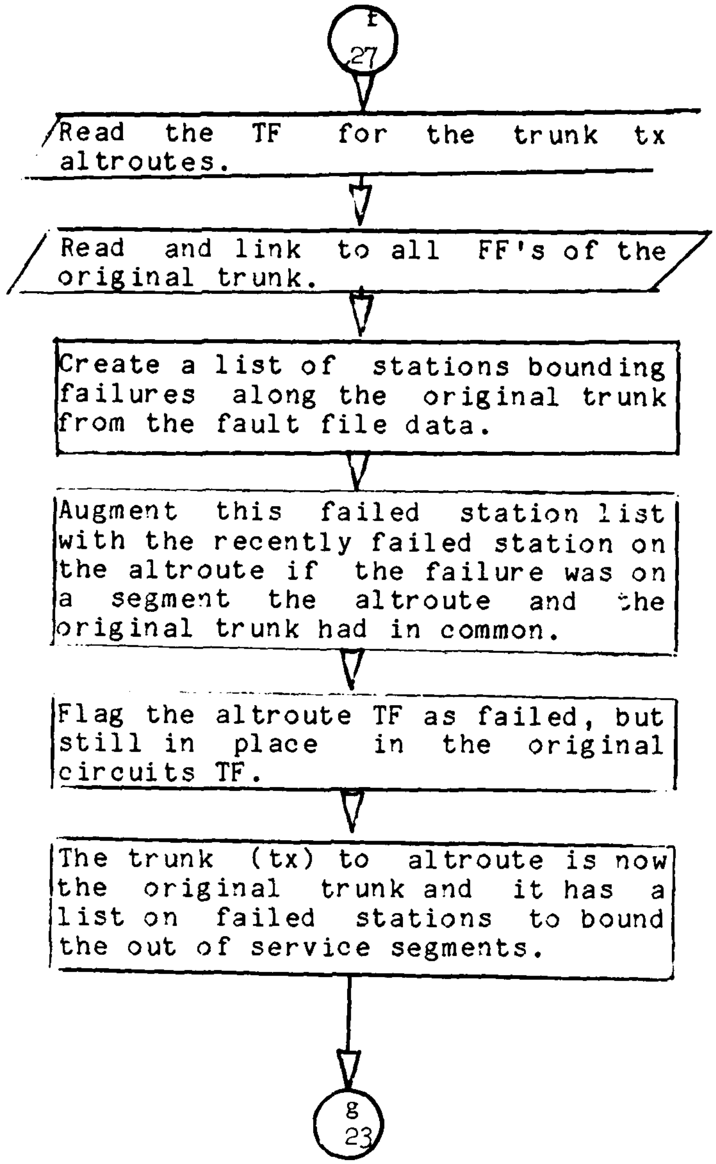
2.6.1.3
Flow Chart of the Main Calling Routine Trunk Section

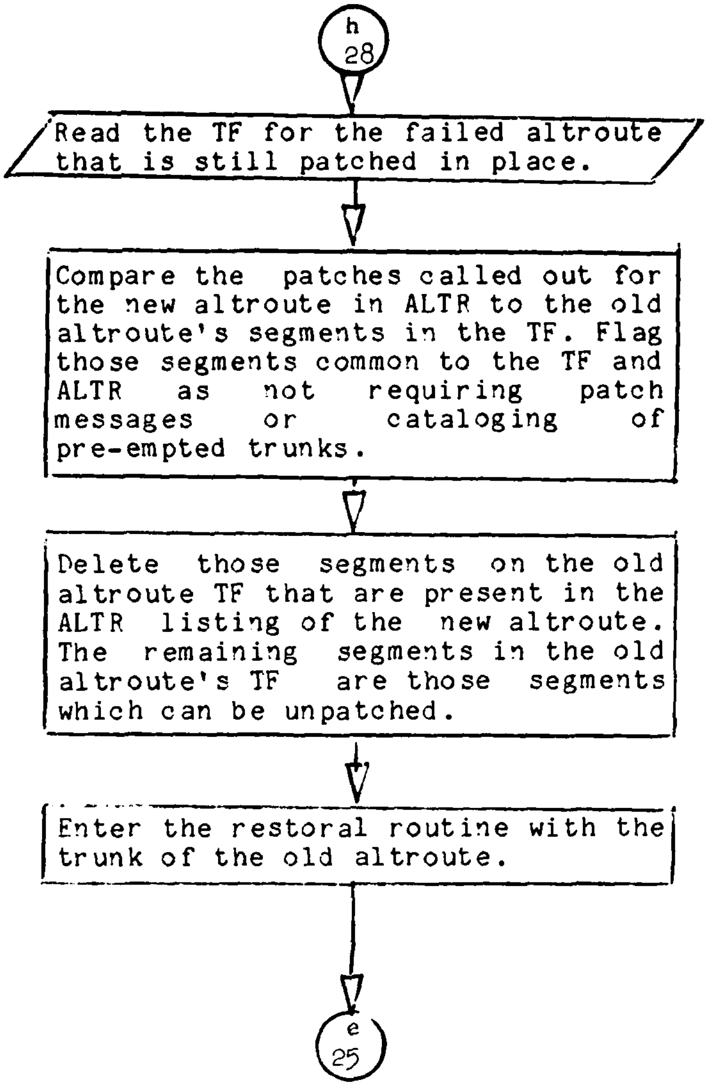



a
24
Dump the TF for tx.
d
24
Is the catalog of trunks to altroute empty?
no
yes
Select the trunk (t) at the top of the altroute catalog (it has the largest sum of RP's).
Go to the circuit altroute catalog and altroute at a circuit level.
j
30
Enter the trunk level altrouting routine.
b
25
yes
Success?
no
c
26
yes
FS=OS?
no
Label FS as a failed station & let the new FS be the next station towards OS with access to T.

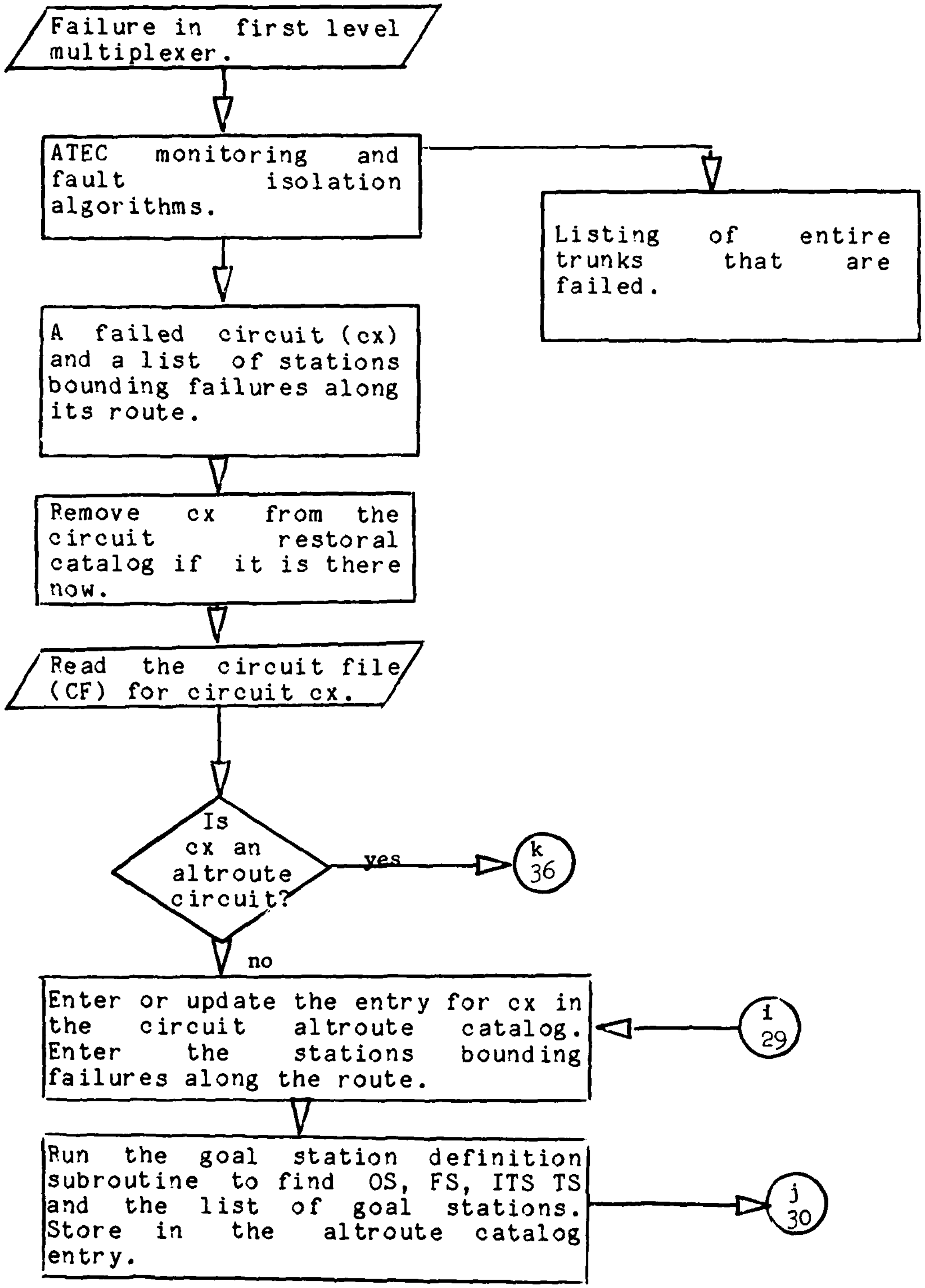

b 25
Is file ALTR blank?
yes
no
Does t have a failed alt-route in place now?
yes
h 28
No
e 25
No need to find an altroute for this trunk. Drop it from the altroute catalog.
Notify users of trunks to be pre-empted.
Catalog trunks that were pre-empted by sum of their RP's in the catalog of trunks to altroute and the catalog of trunks to restore. The stations bounding pre-empted segments of the pre-empted trunks are listed as bounding failures in the altroute catalog entry.
Direct stations to patch in altroutes.
Create new TF for the altroute & provide pointers in appropriate TF's and LF's.
Drop T from the catalog of trunks to altroute.
d 24

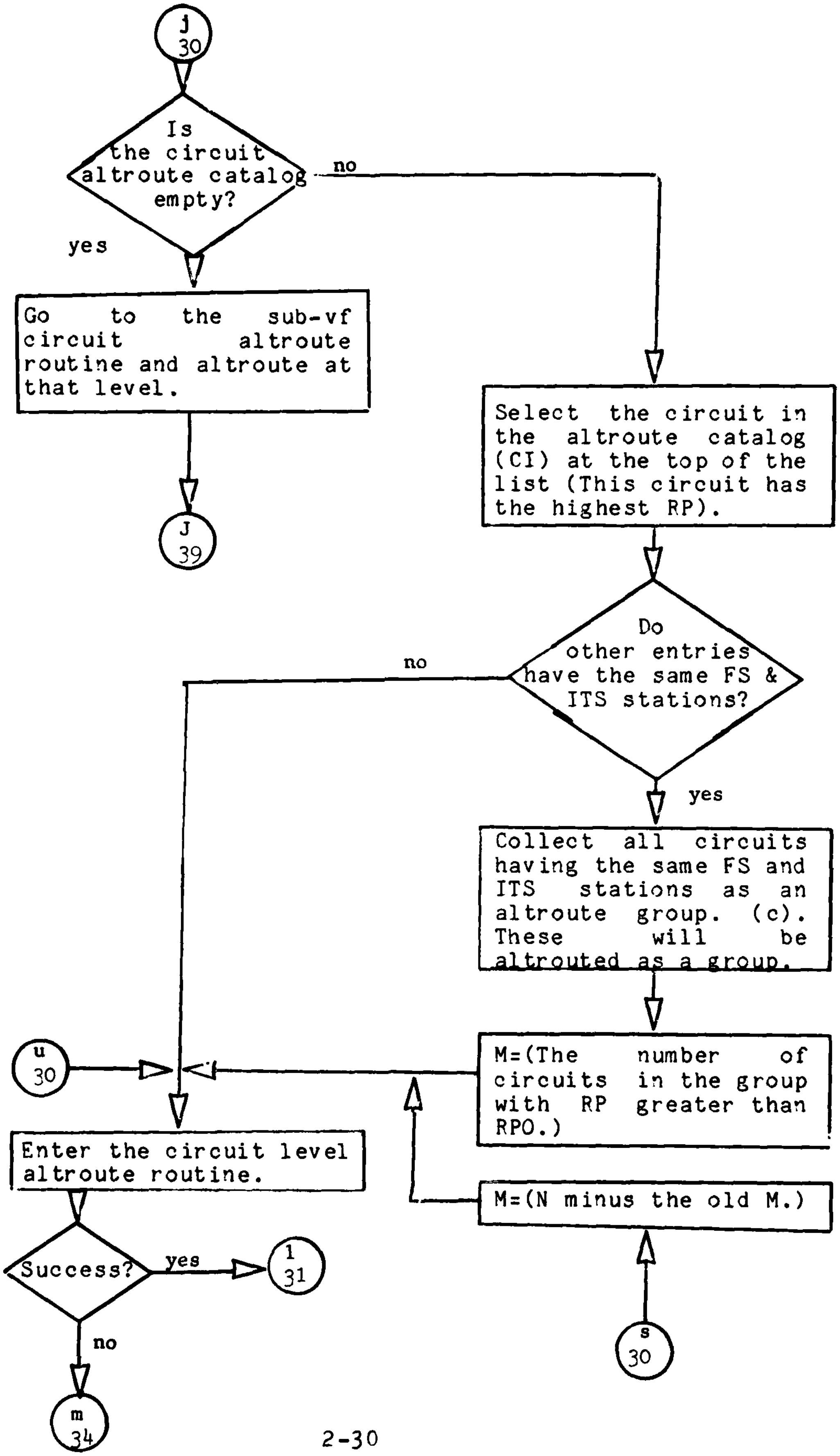

c
26
Enter the circuits carried by the trunk into the circuit altroute catalog by RP ordering. List the stations bounding failures on the trunk with each entry as stations bounding the circuit failures.
If the trunk was on the restoral catalog, place the circuits on the circuit restoral catalog.
Transfer any pre-emptions of the trunk to pre-emptions in the CF's of the carried circuits.
Run the circuit goal station definition routine for each circuit on the trunk and enter the results into a circuit altroute catalog entry for each circuit.
Do not keep a circuit altroute entry for circuits which fail the goal station definition search. They are isolated on a spur and cannot be altrouted. Delete T from the catalog of trunks to altroute.
Remove T from the trunk altroute catalog and restoral catalog.
d
24

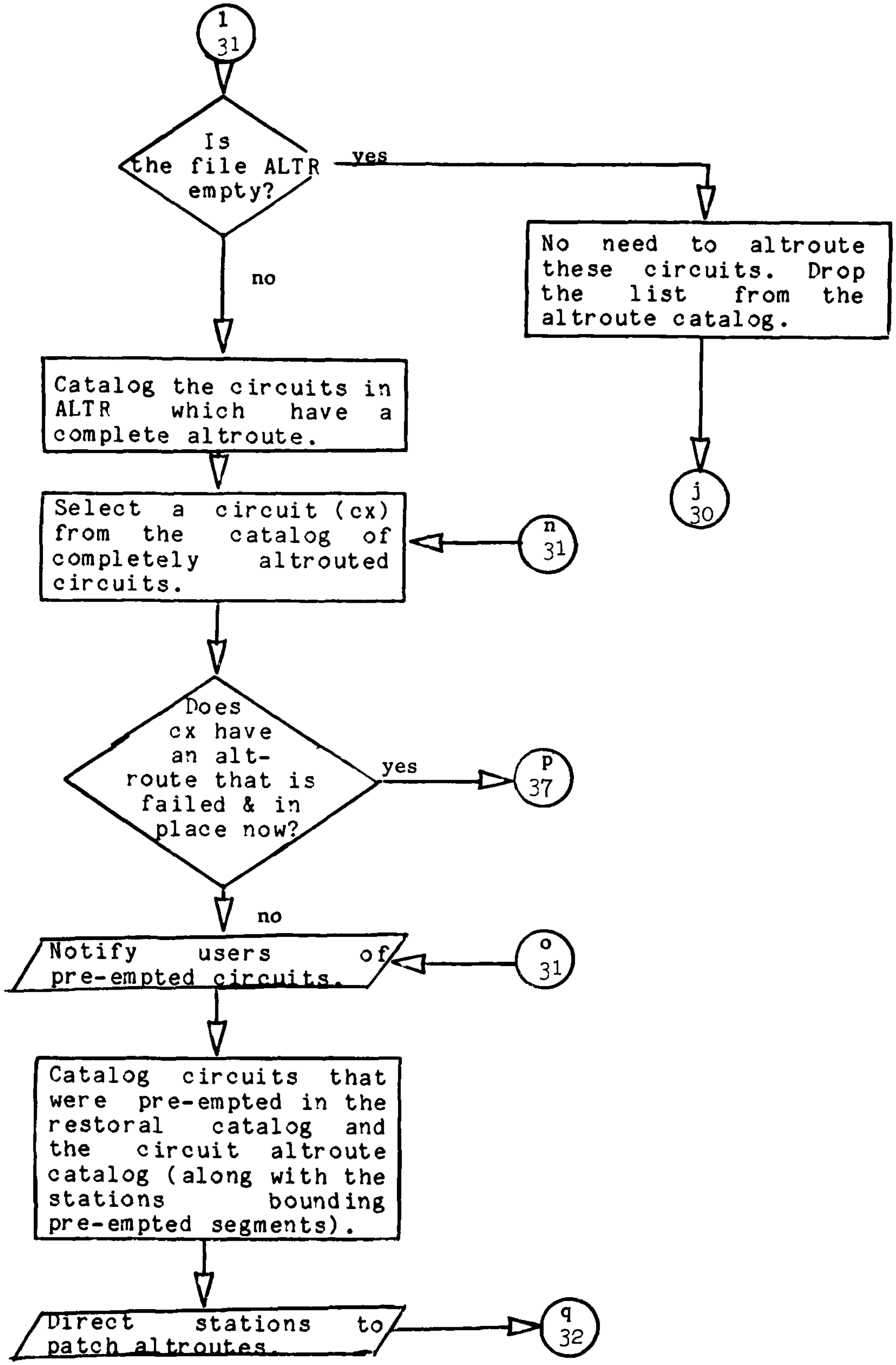

f
27
Read the TF for the trunk tx altroutes.
Read and link to all FF's of the original trunk.
Create a list of stations bounding failures along the original trunk from the fault file data.
Augment this failed station list with the recently failed station on the altroute if the failure was on a segment the altroute and the original trunk had in common.
Flag the altroute TF as failed, but still in place in the original circuits TF.
The trunk (tx) to altroute is now the original trunk and it has a list on failed stations to bound the out of service segments.
g
23

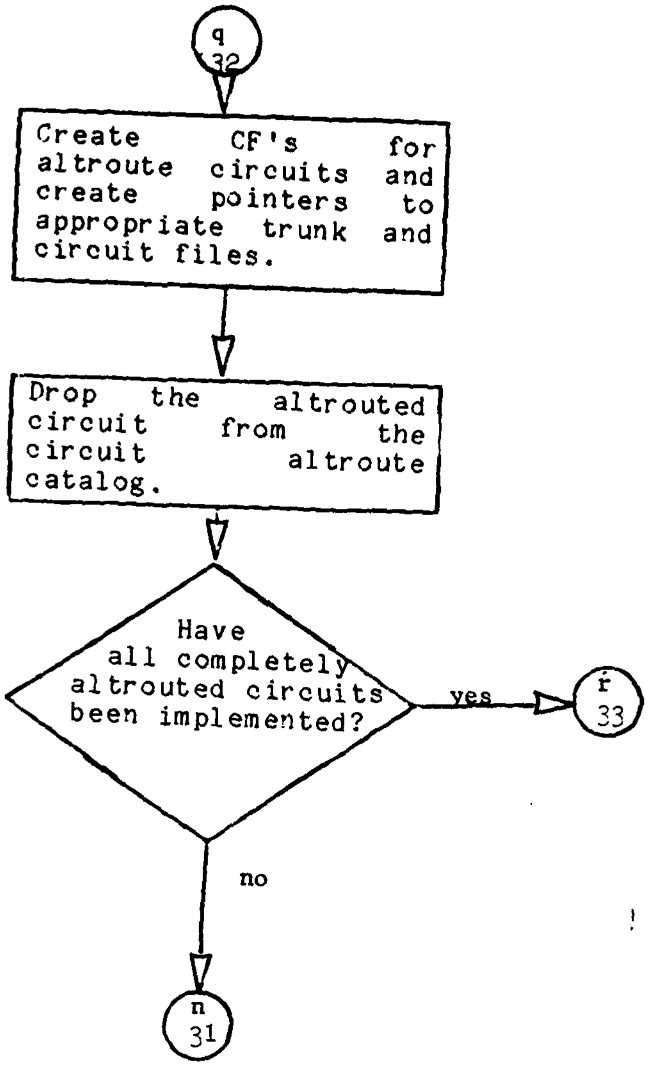

h
28
Read the TF for the failed altroute that is still patched in place.
Compare the patches called out for the new altroute in ALTR to the old altroute's segments in the TF. Flag those segments common to the TF and ALTR as not requiring patch messages or cataloging of pre-empted trunks.
Delete those segments on the old altroute TF that are present in the ALTR listing of the new altroute. The remaining segments in the old altroute's TF are those segments which can be unpatched.
Enter the restoral routine with the trunk of the old altroute.
e
25

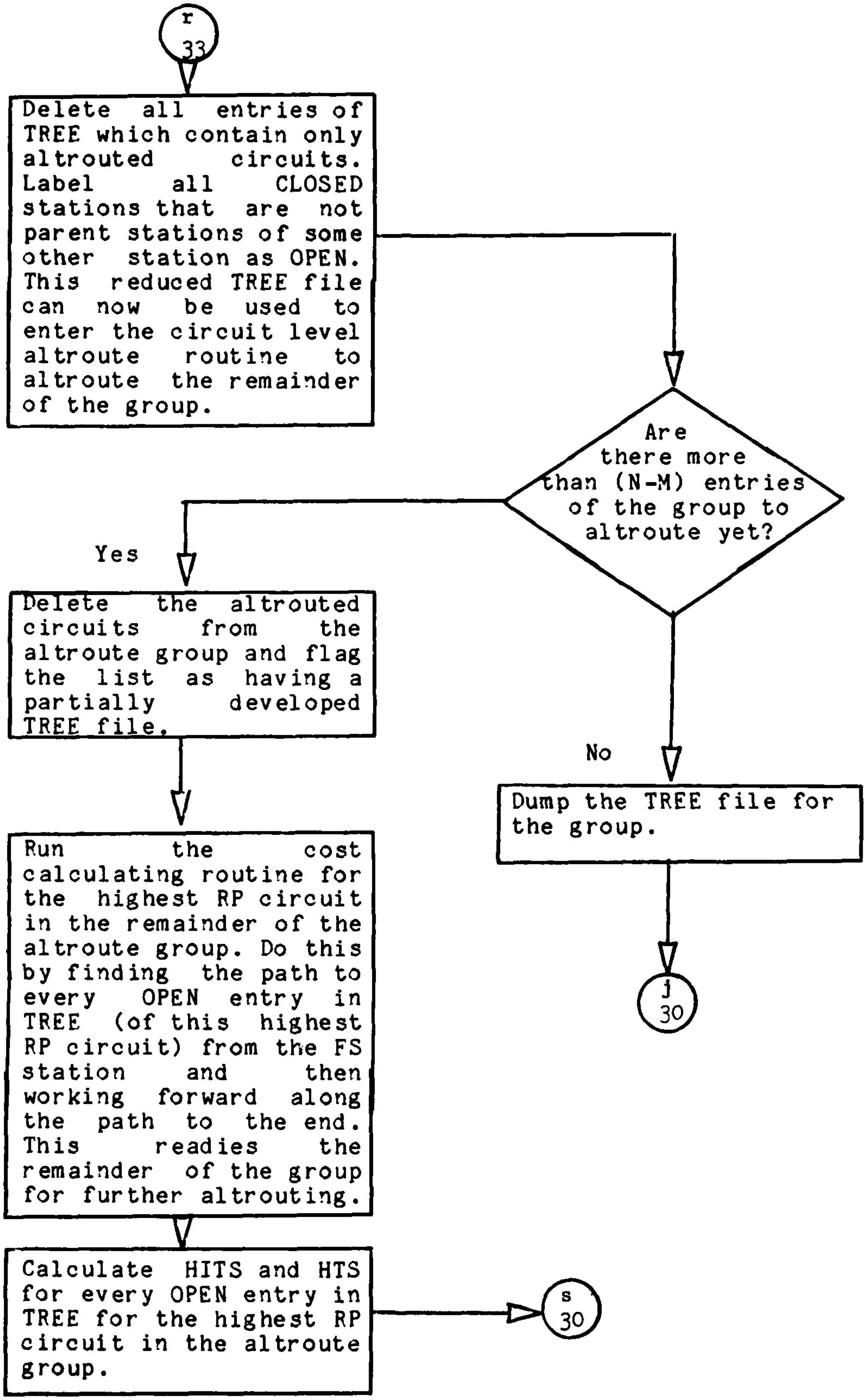
Circuit Section

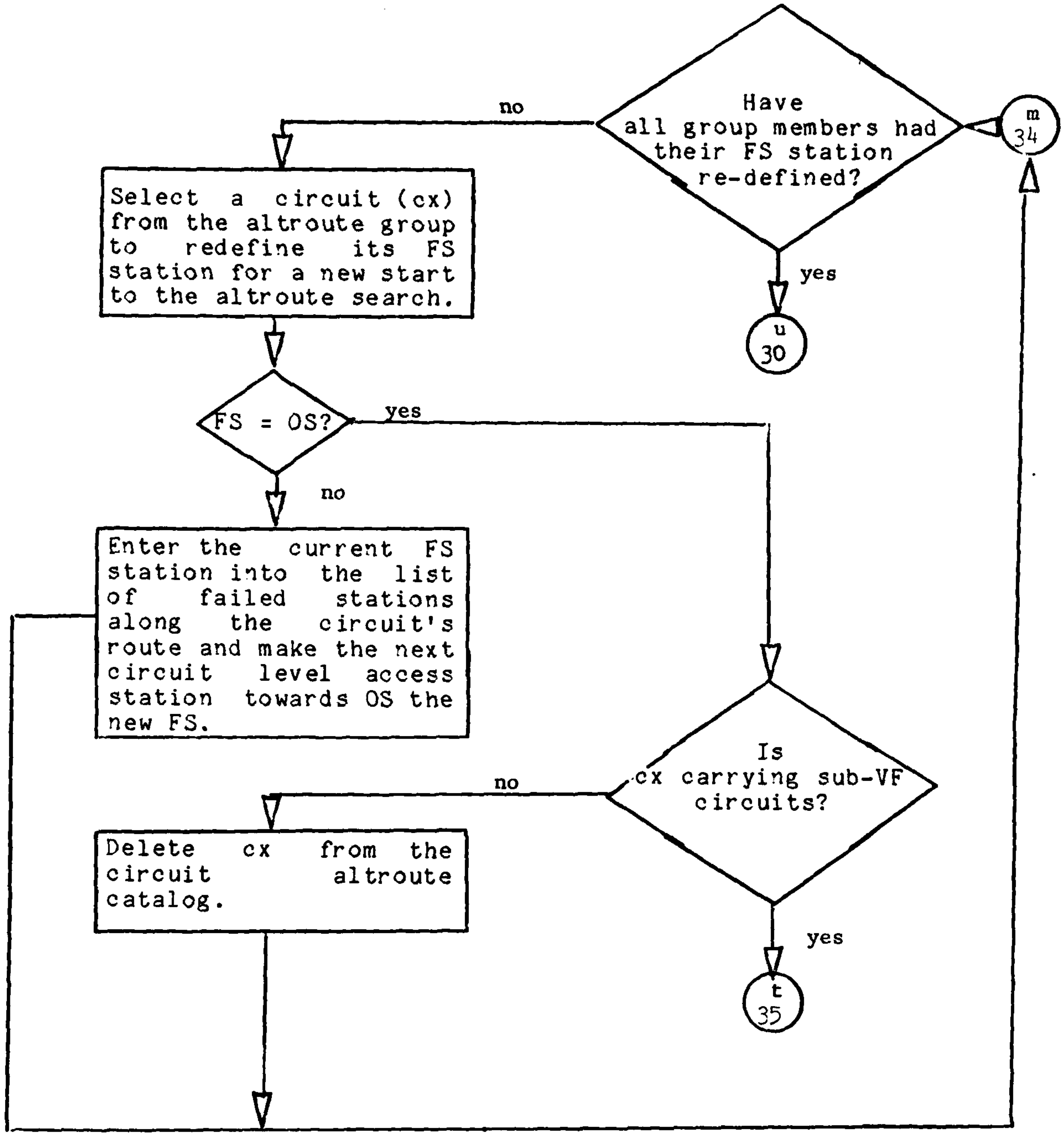



j 30
Is the circuit altroute catalog empty?
no
yes
Go to the sub-vf circuit altroute routine and altroute at that level.
J 39
Select the circuit in the altroute catalog (CI) at the top of the list (This circuit has the highest RP).
Do other entries have the same FS & ITS stations?
no
yes
Collect all circuits having the same FS and ITS stations as an altroute group. (c). These will be altrouted as a group.
M=(The number of circuits in the group with RP greater than RPO.)
u 30
Enter the circuit level altroute routine.
M=(N minus the old M.)
Success?
yes
1 31
no
m 34
s 30


1 31
Is the file ALTR empty?
yes
No need to altroute these circuits. Drop the list from the altroute catalog.
j 30
no
Catalog the circuits in ALTR which have a complete altroute.
Select a circuit (cx) from the catalog of completely altrouted circuits.
n 31
Does cx have an alt-route that is failed & in place now?
yes
p 37
no
Notify users of pre-empted circuits.
o 31
Catalog circuits that were pre-empted in the restoral catalog and the circuit altroute catalog (along with the stations bounding pre-empted segments).
Direct stations to patch altroutes.
q 32


q
32
Create CF's for altroute circuits and create pointers to appropriate trunk and circuit files.
Drop the altrouted circuit from the circuit altroute catalog.
Have all completely altrouted circuits been implemented?
yes
r
33
no
n
31


r
33
Delete all entries of TREE which contain only altrouted circuits. Label all CLOSED stations that are not parent stations of some other station as OPEN. This reduced TREE file can now be used to enter the circuit level altroute routine to altroute the remainder of the group.
Are there more than (N-M) entries of the group to altroute yet?
Yes
Delete the altrouted circuits from the altroute group and flag the list as having a partially developed TREE file.
Run the cost calculating routine for the highest RP circuit in the remainder of the altroute group. Do this by finding the path to every OPEN entry in TREE (of this highest RP circuit) from the FS station and then working forward along the path to the end. This readies the remainder of the group for further altrouting.
Calculate HITS and HTS for every OPEN entry in TREE for the highest RP circuit in the altroute group.
s
30
No
Dump the TREE file for the group.
j
30


Have all group members had their FS station re-defined?
no
yes
m
34
u
30
Select a circuit (cx) from the altroute group to redefine its FS station for a new start to the altroute search.
FS = OS?
yes
no
Enter the current FS station into the list of failed stations along the circuit's route and make the next circuit level access station towards OS the new FS.
Is cx carrying sub-VF circuits?
no
yes
Delete cx from the circuit altroute catalog.
t
35

t
35

Enter the circuits into the catalog of sub-vf circuits to altroute. Enter the sub-vf circuits carried by the vf circuit into the catalog of sub-vf circuits to altroute. Order them by RP. List the station ends of the carrying circuit as stations bounding the failures on the sub-vf circuit.

Place the VF circuits and sub-vf circuits on the circuit restoral catalog.

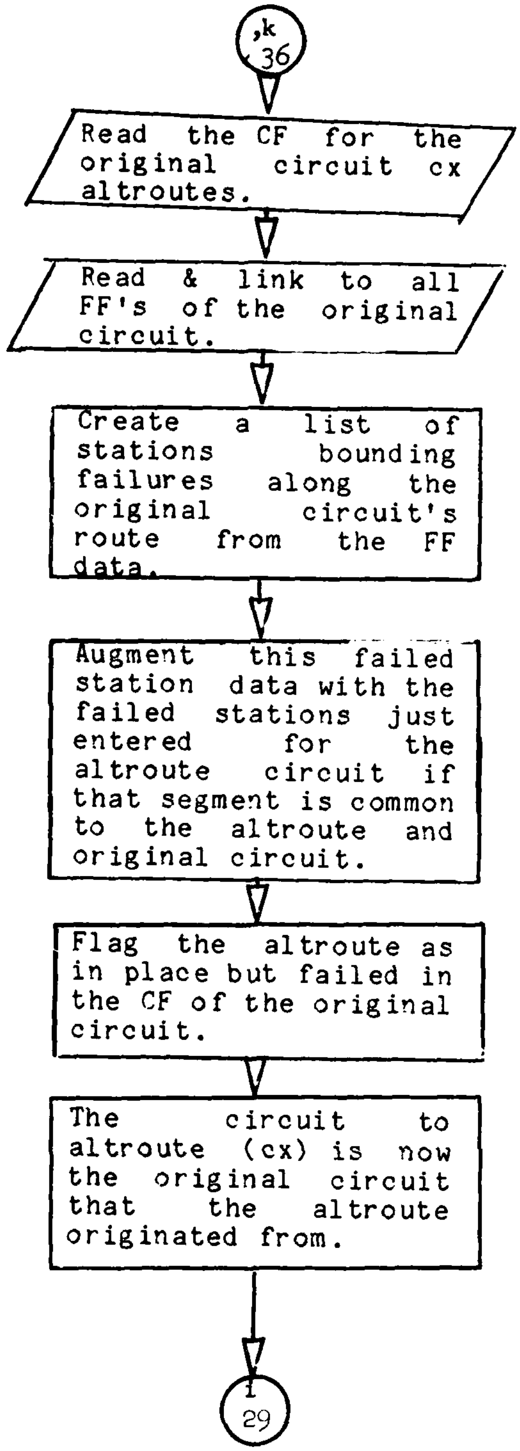
Carry over any pre-emptions on the vf circuit to the sub-vf circuits being carried.

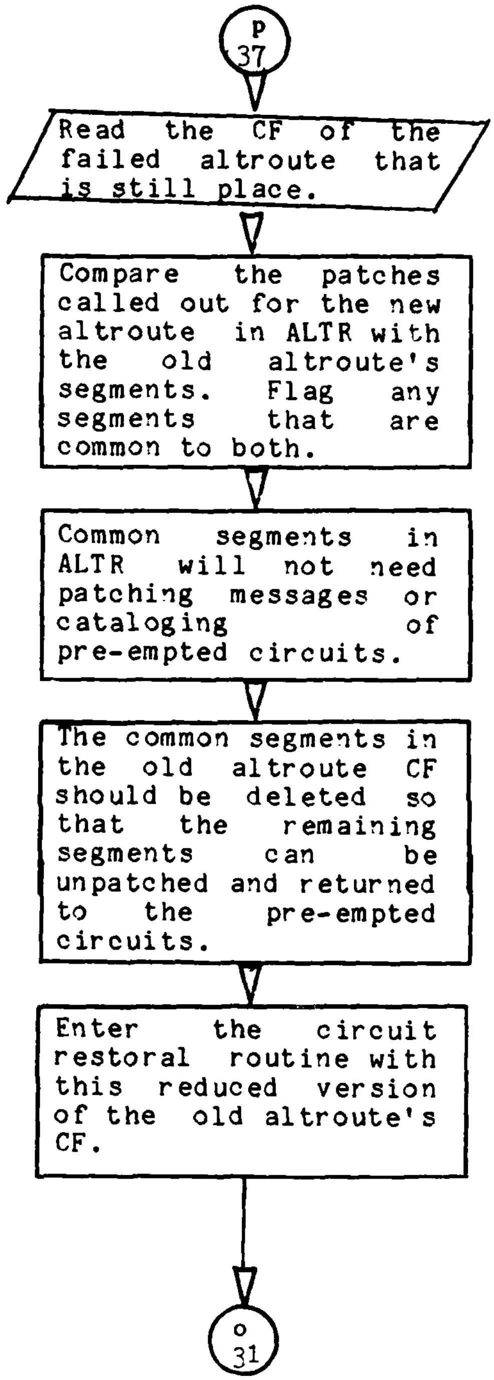
Run the sub-vf circuit goal station definition routine to find the OS, FS, ITS, TS and goal stations of the circuit.

Remove cx from the circuit altroute and restoral catalogs.

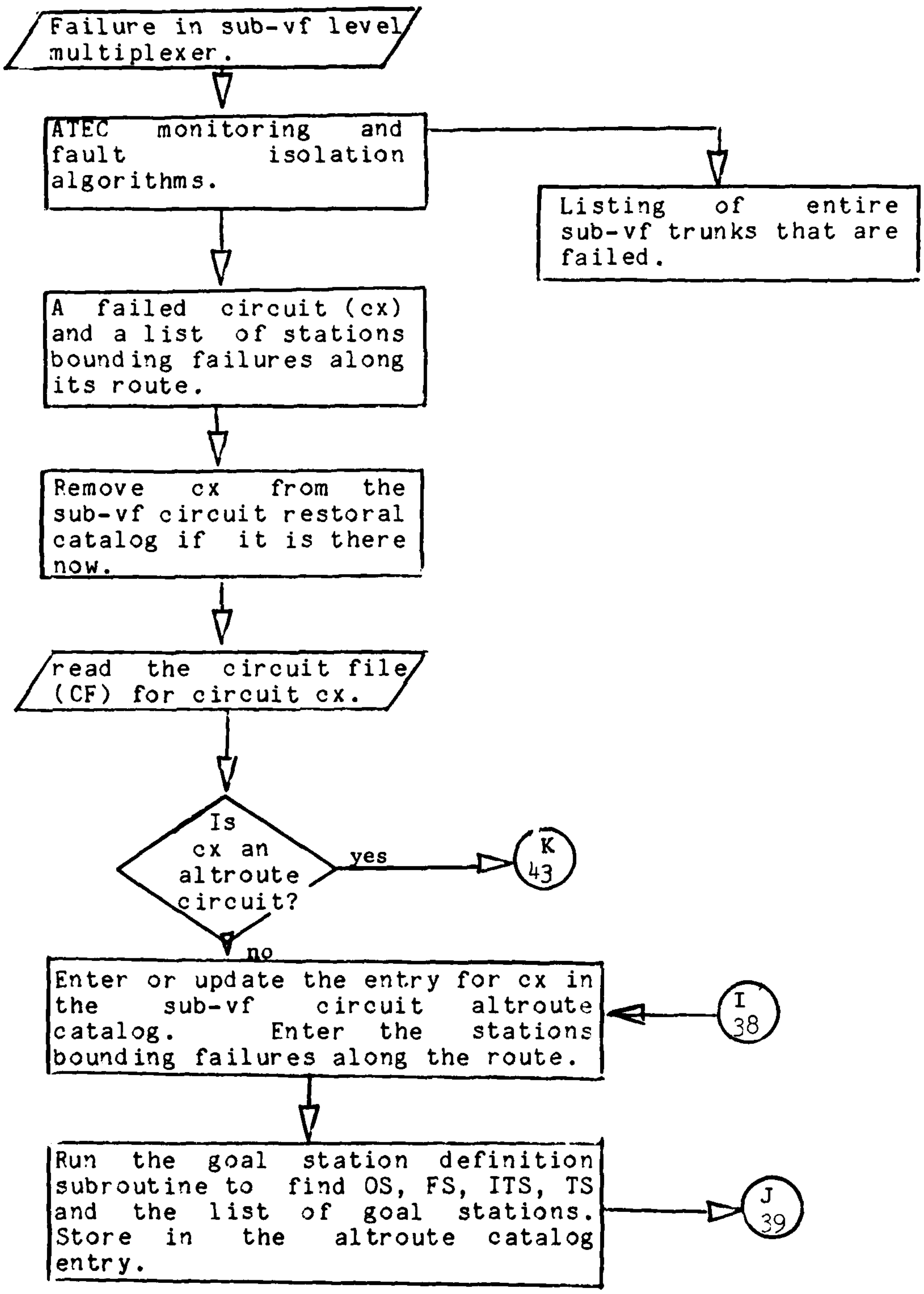
Do not keep a circuit which leaves the goal station definition routine in failure. That circuit is isolated on a spur and cannot be altrouted.

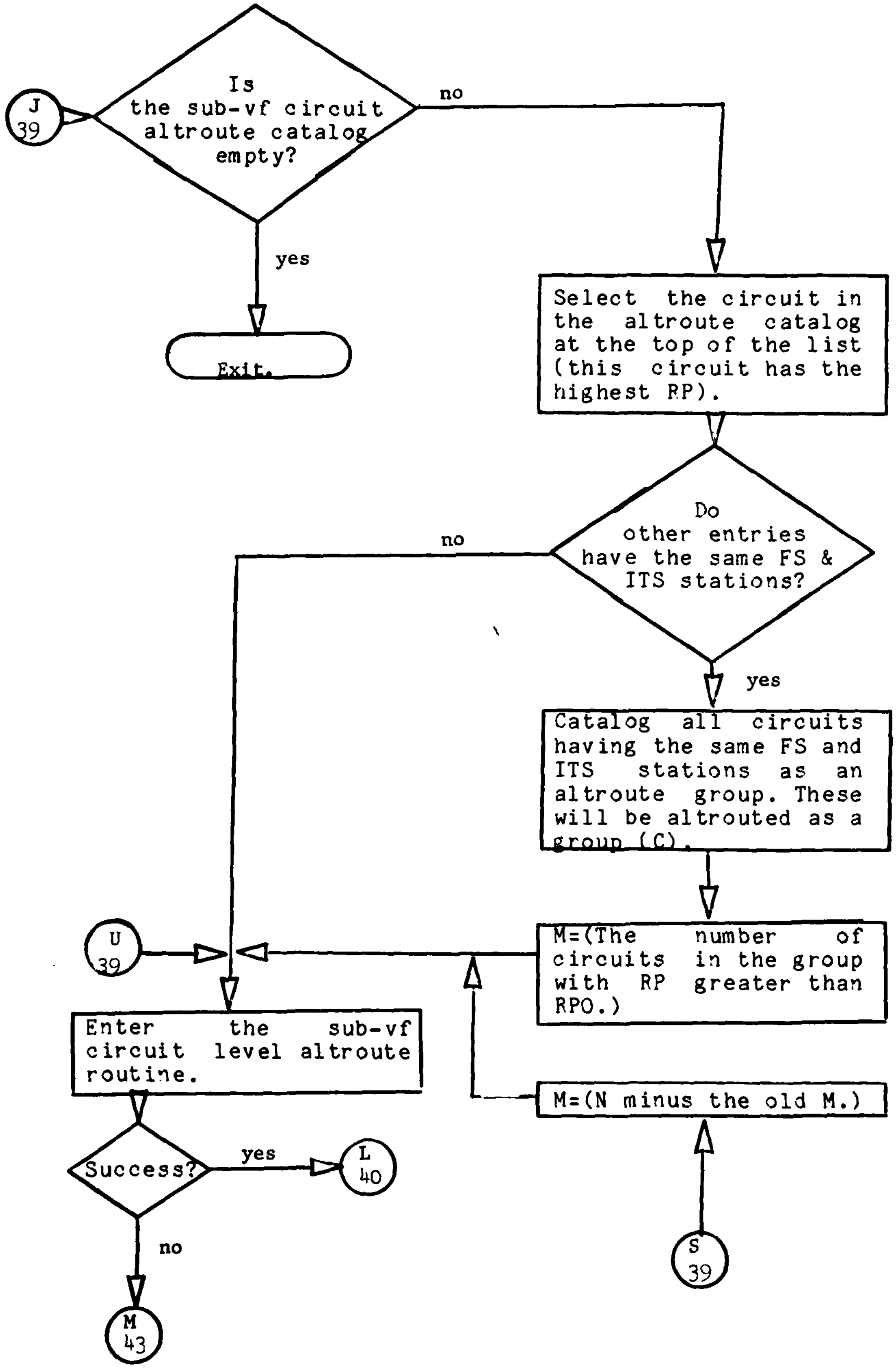
m
34

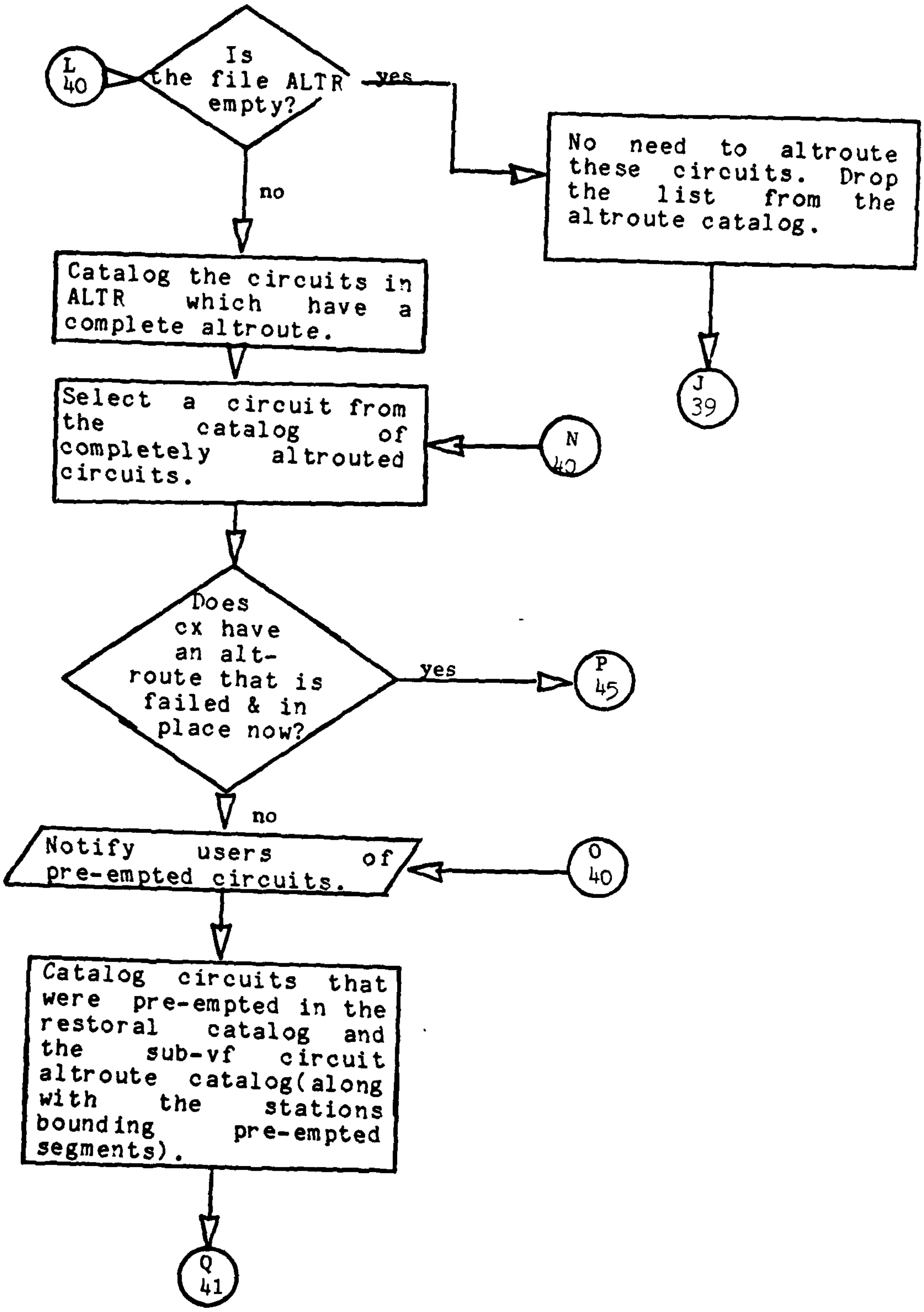

k
36
Read the CF for the original circuit cx altroutes.
Read & link to all FF's of the original circuit.
Create a list of stations bounding failures along the original circuit's route from the FF data.
Augment this failed station data with the failed stations just entered for the altroute circuit if that segment is common to the altroute and original circuit.
Flag the altroute as in place but failed in the CF of the original circuit.
The circuit to altroute (cx) is now the original circuit that the altroute originated from.
I
29

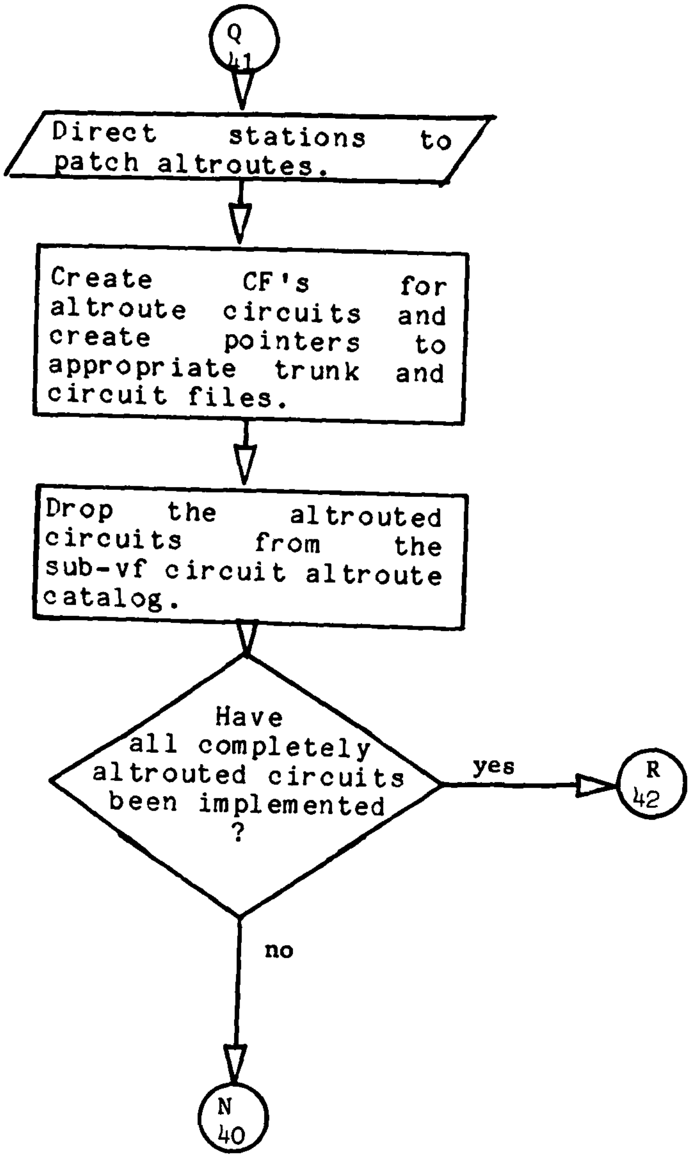

P
37
Read the CF of the failed altroute that is still place.
Compare the patches called out for the new altroute in ALTR with the old altroute's segments. Flag any segments that are common to both.
Common segments in ALTR will not need patching messages or cataloging of pre-empted circuits.
The common segments in the old altroute CF should be deleted so that the remaining segments can be unpatched and returned to the pre-empted circuits.
Enter the circuit restoral routine with this reduced version of the old altroute's CF.
o
31

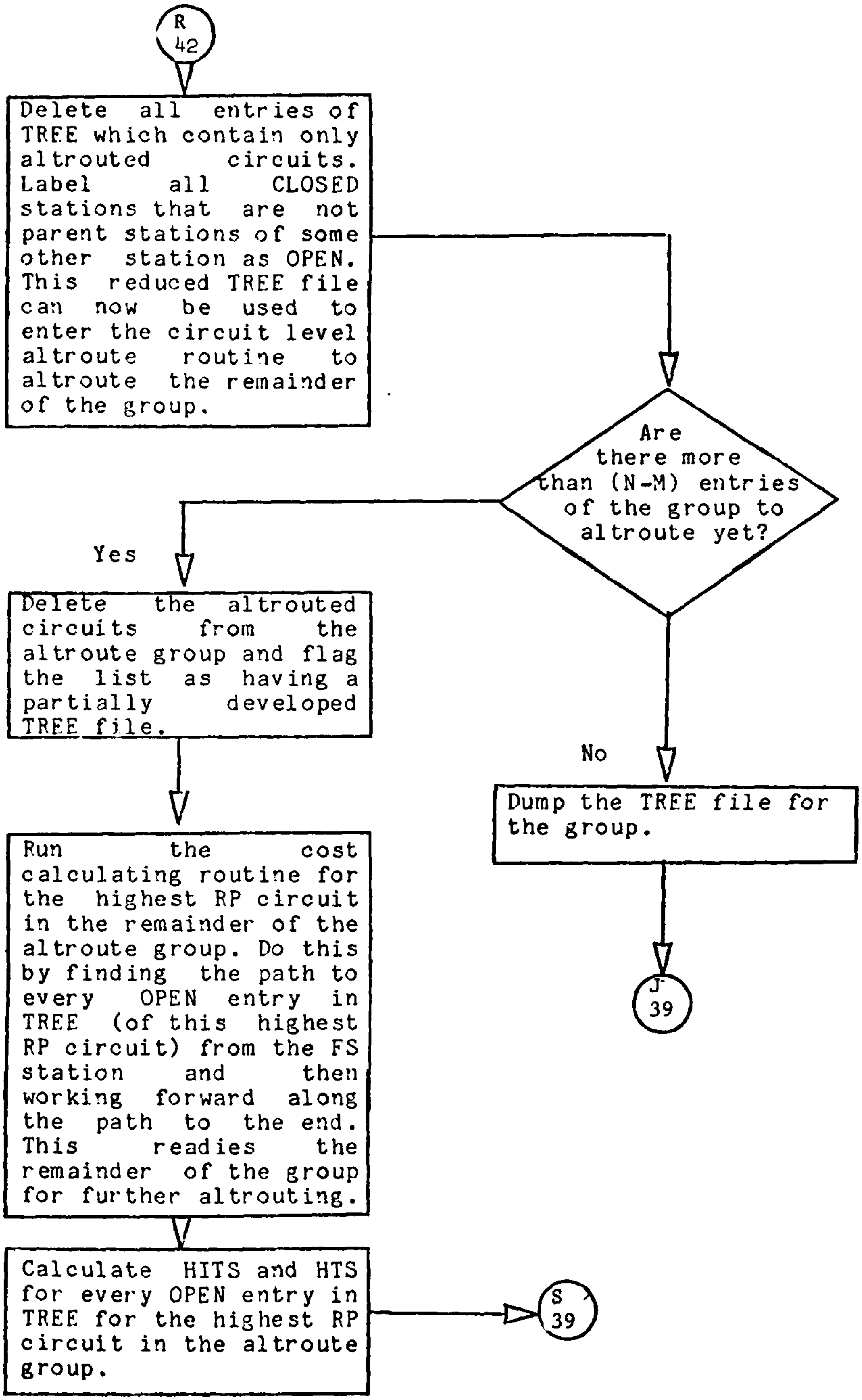
Sub-vf Circuit Section

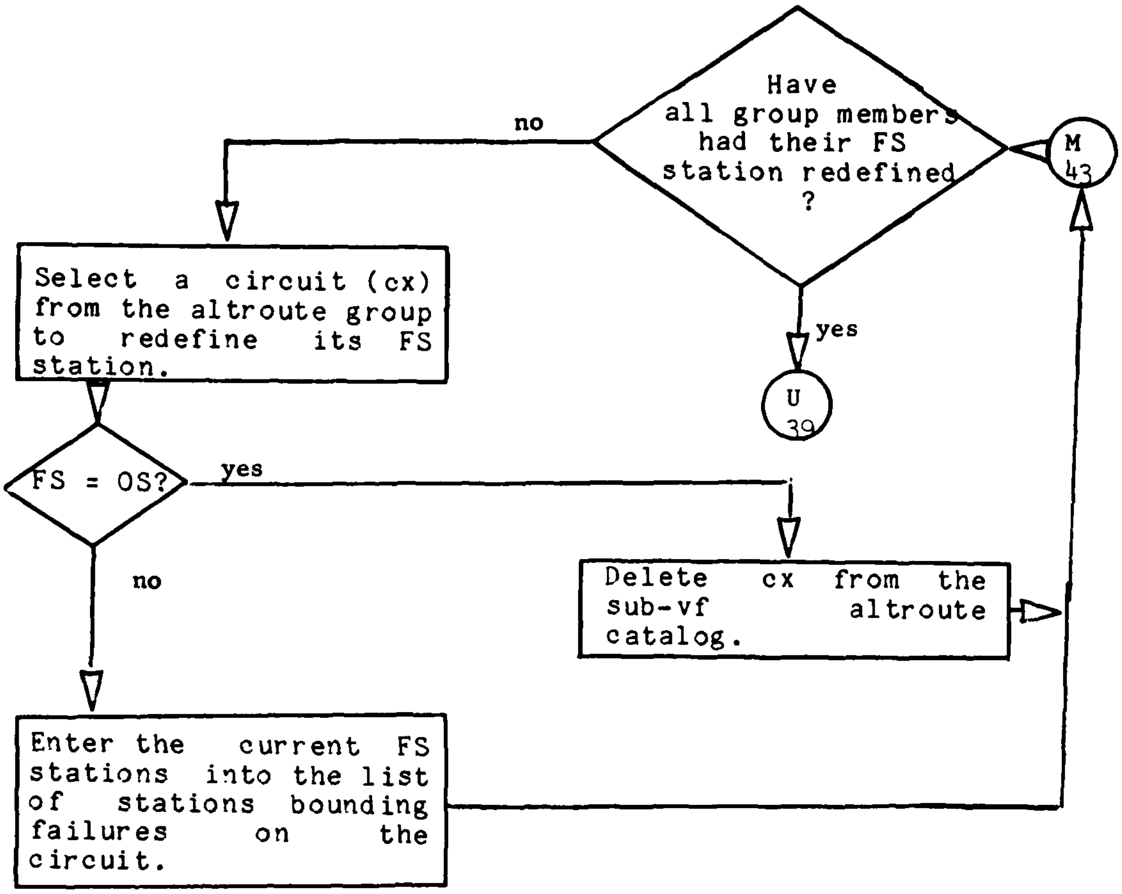


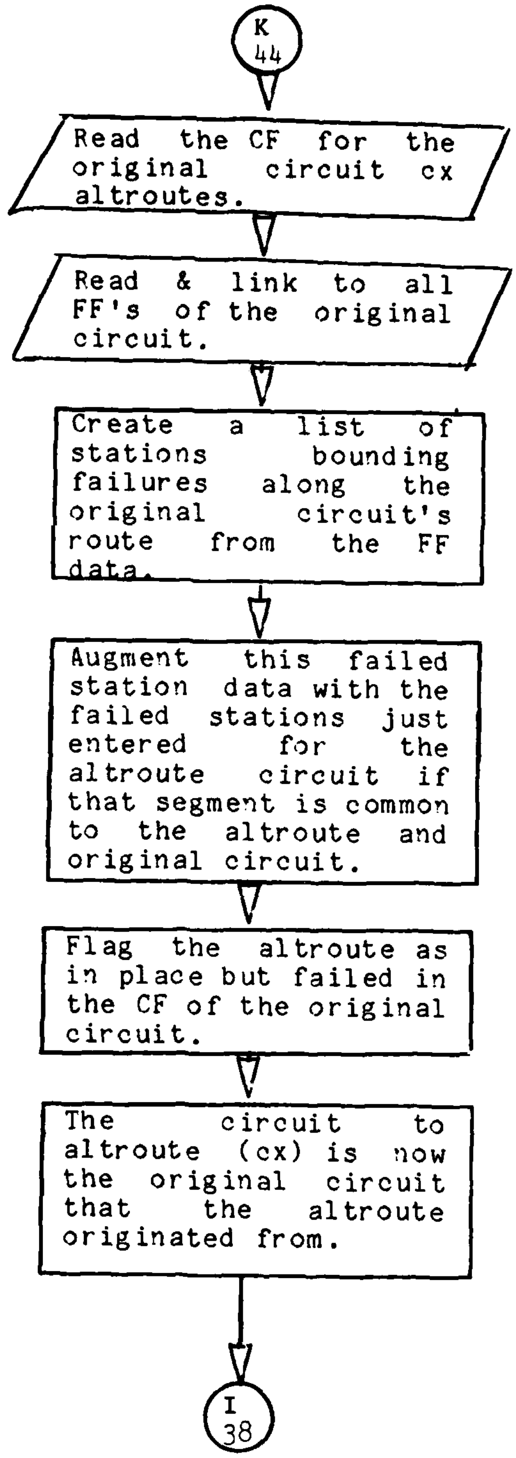

J 39
Is the sub-vf circuit altroute catalog empty?
no
yes
Exit.
Select the circuit in the altroute catalog at the top of the list (this circuit has the highest RP).
Do other entries have the same FS & ITS stations?
no
yes
Catalog all circuits having the same FS and ITS stations as an altroute group. These will be altrouted as a group (C).
U 39
M=(The number of circuits in the group with RP greater than RPO.)
Enter the sub-vf circuit level altroute routine.
M=(N minus the old M.)
Success?
yes
L 40
no
M 43
S 39

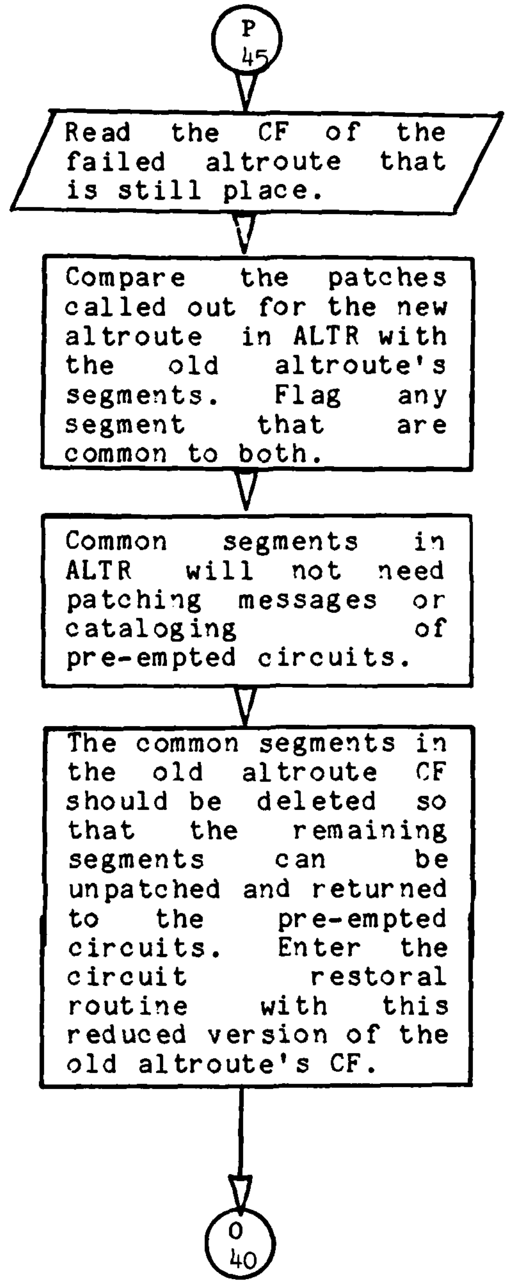

L 40
Is the file ALTR empty?
yes
No need to altroute these circuits. Drop the list from the altroute catalog.
J 39
no
Catalog the circuits in ALTR which have a complete altroute.
Select a circuit from the catalog of completely altrouted circuits.
N 40
Does cx have an alt-route that is failed & in place now?
yes
P 45
no
Notify users of pre-empted circuits.
O 40
Catalog circuits that were pre-empted in the restoral catalog and the sub-vf circuit altroute catalog(along with the stations bounding pre-empted segments).
Q 41


Q
41
Direct stations to patch altroutes.
Create CF's for altroute circuits and create pointers to appropriate trunk and circuit files.
Drop the altrouted circuits from the sub-vf circuit altroute catalog.
Have all completely altrouted circuits been implemented ?
yes
R
42
no
N
40


R 42
Delete all entries of TREE which contain only altrouted circuits. Label all CLOSED stations that are not parent stations of some other station as OPEN. This reduced TREE file can now be used to enter the circuit level altroute routine to altroute the remainder of the group.
Are there more than (N-M) entries of the group to altroute yet?
Yes
Delete the altrouted circuits from the altroute group and flag the list as having a partially developed TREE file.
Run the cost calculating routine for the highest RP circuit in the remainder of the altroute group. Do this by finding the path to every OPEN entry in TREE (of this highest RP circuit) from the FS station and then working forward along the path to the end. This readies the remainder of the group for further altrouting.
Calculate HITS and HTS for every OPEN entry in TREE for the highest RP circuit in the altroute group.
S 39
No
Dump the TREE file for the group.
J 39


Have all group members had their FS station redefined ?
no
yes
M 43
U 39
Select a circuit (cx) from the altroute group to redefine its FS station.
FS = OS?
yes
no
Delete cx from the sub-vf altroute catalog.
Enter the current FS stations into the list of stations bounding failures on the circuit.


K 44
Read the CF for the original circuit cx altroutes.
Read & link to all FF's of the original circuit.
Create a list of stations bounding failures along the original circuit's route from the FF data.
Augment this failed station data with the failed stations just entered for the altroute circuit if that segment is common to the altroute and original circuit.
Flag the altroute as in place but failed in the CF of the original circuit.
The circuit to altroute (cx) is now the original circuit that the altroute originated from.
I 38


P
45
Read the CF of the failed altroute that is still place.
Compare the patches called out for the new altroute in ALTR with the old altroute's segments. Flag any segment that are common to both.
Common segments in ALTR will not need patching messages or cataloging of pre-empted circuits.
The common segments in the old altroute CF should be deleted so that the remaining segments can be unpatched and returned to the pre-empted circuits. Enter the circuit restoral routine with this reduced version of the old altroute's CF.
O
40

2.6.2 The Trunk Altrouting Routine

2.6.2.1 Discussion of the Routine and Simplified Flow Chart-- This routine is called whenever an entire trunk has failed. We attempt to altroute the trunk at the group level of the multiplexer hierarchy in this routine. Only spare groups or groups carrying trunks whose RP's are all lower than the RP's of the trunk to altroute are possible for altrouting.

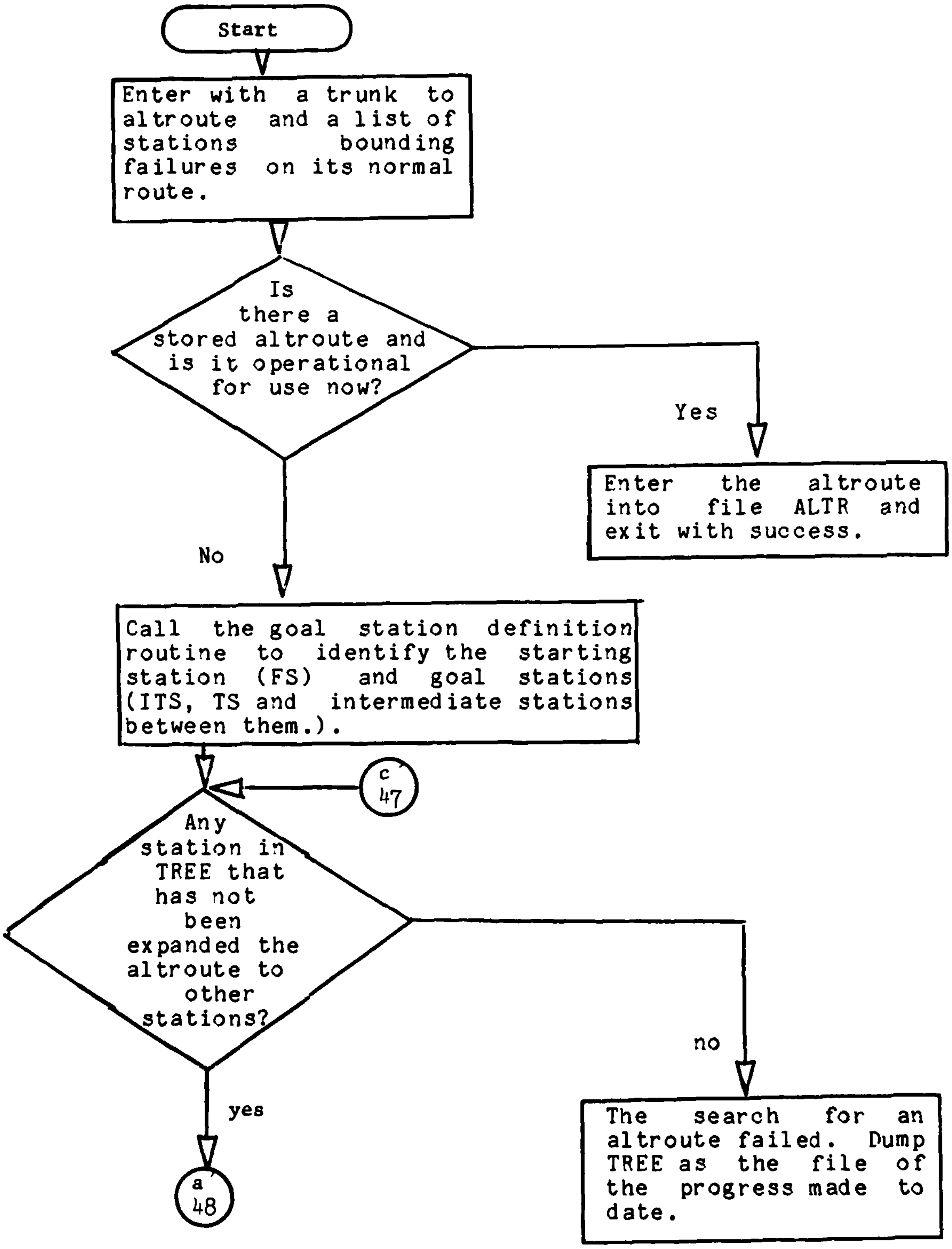
Between the simplified flow chart in this section and the detailed flow chart in section 2.6.2.3, the routine is well defined. Only a few points need be made here to clarify the routine:

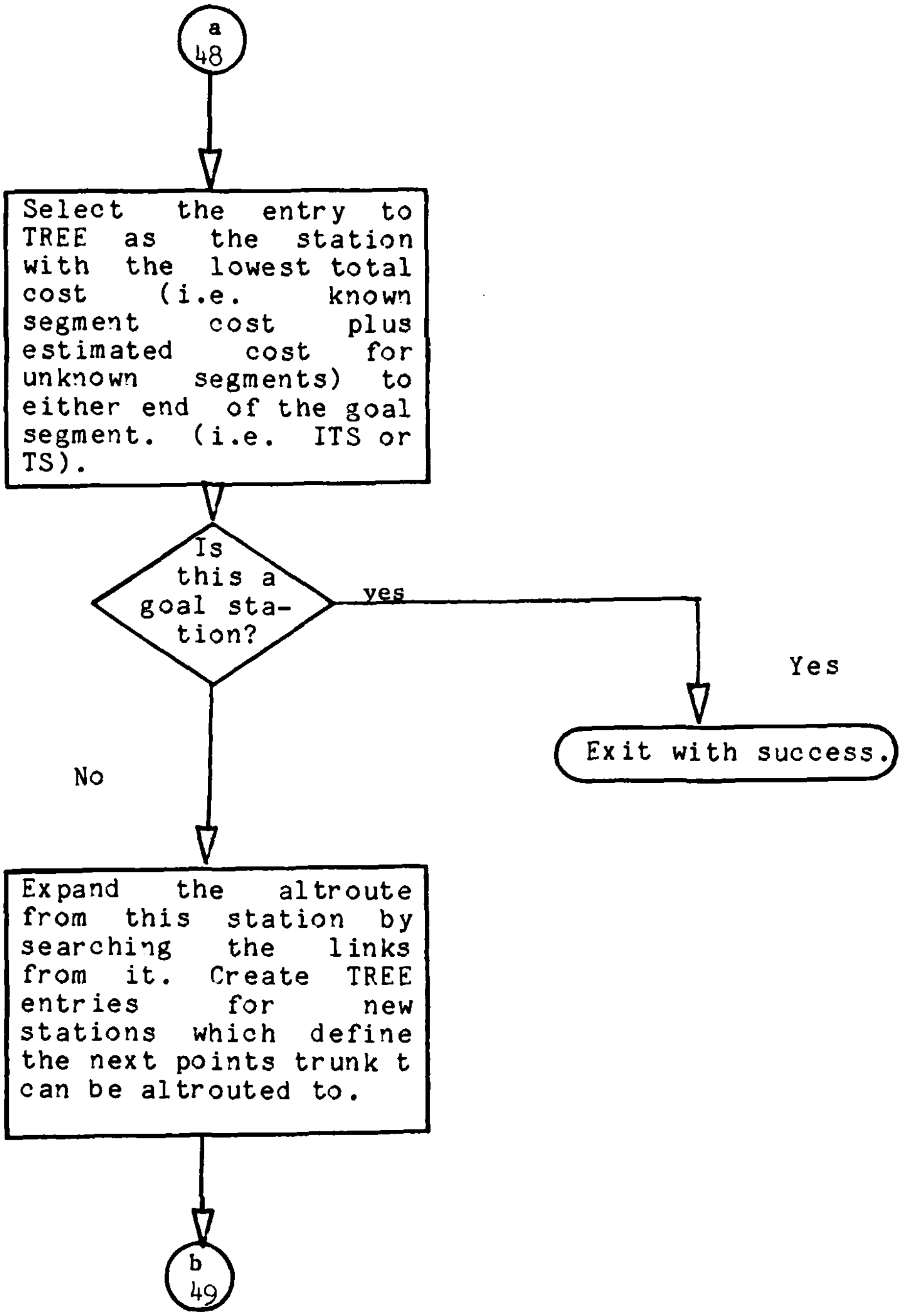
(1) "Trunks" always have exactly 24 circuits riding them when they are non-satellite links. The digitization of the DCS will, therefore insure that one trunk's capacity can be placed on another pre-empted trunk or spare group for non-satellite links. Satellite links do not now have this property. If they do have trunk capacity compatibility in the future, then they can be entered into the trunk files just like surface trunks and be handled the same. The routine assumes that this is the case. In the event satellite links do not have compatability with surface trunks, then they should not be considered in the trunk level altrouting. The whole purpose of this routine is to find the altroute with the minimum of decomposition. If an altroute is not found due to this omission, then it would be found when the main routine decomposes the trunk into circuits and calls for circuit level altrouting.

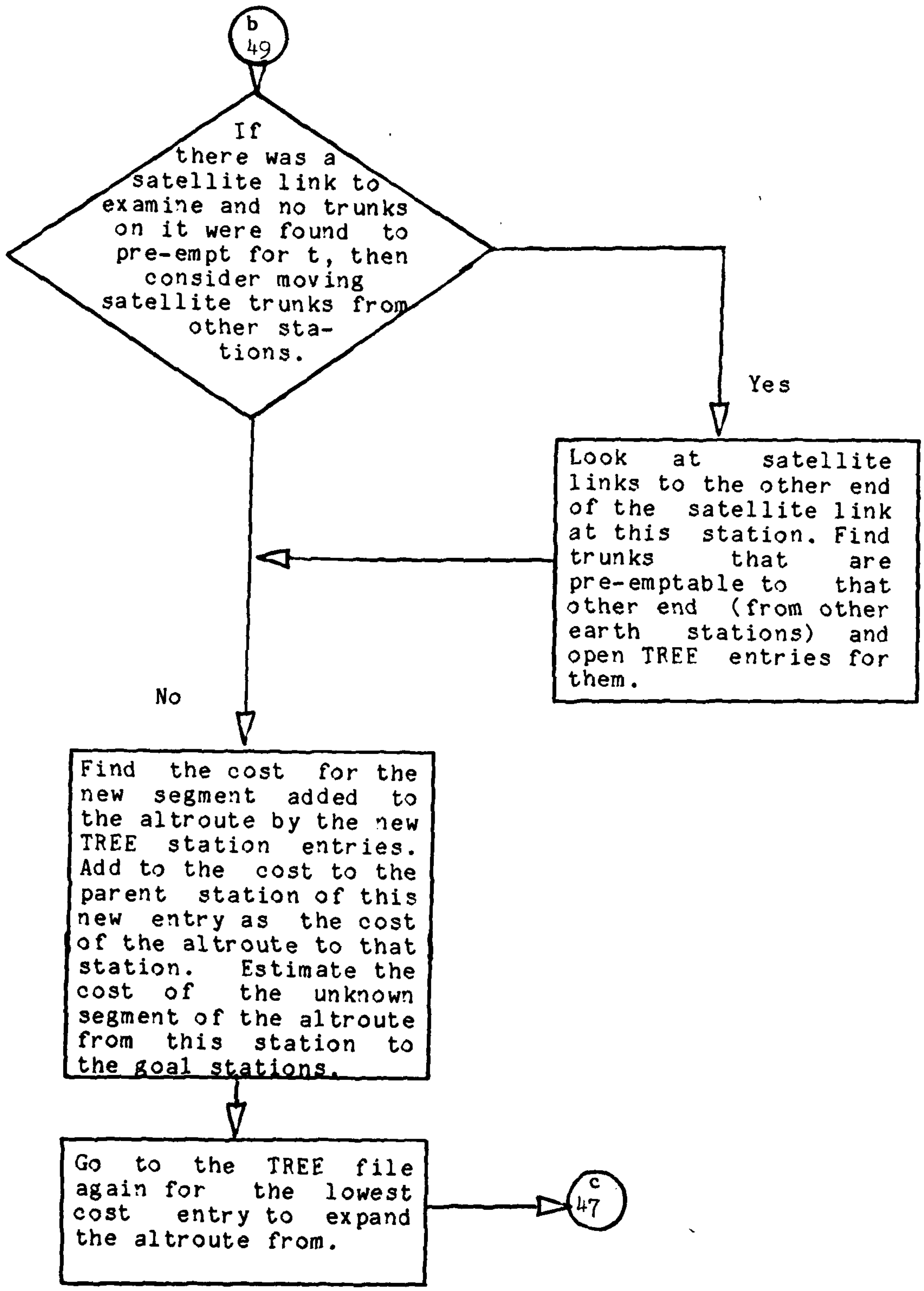
(2) The data storing the partial altroute and all of its possible variations is a file called TREE. Section 2.6.2.2 and Figure 2-2 detail the form of this rather large and important data storage file for the routine. Other data needed for the routine is in the form of simple catalogs of items having some common property and their identification in context is obvious.

(3) Note that the routine must look at all trunks accessible at a station as was discussed earlier. The number of group accessible stations is variable in the DCS trunk network and we need to look at as many branches as are possible from a station to improve the chances of finding an altroute.

Simplified Flow Chart of the Trunk Altrouting Routine




a
48
Select the entry to TREE as the station with the lowest total cost (i.e. known segment cost plus estimated cost for unknown segments) to either end of the goal segment. (i.e. ITS or TS).
Is this a goal station?
yes
Yes
Exit with success.
No
Expand the altroute from this station by searching the links from it. Create TREE entries for new stations which define the next points trunk t can be altrouted to.
b
49


b
49
If there was a satellite link to examine and no trunks on it were found to pre-empt for t, then consider moving satellite trunks from other stations.
Yes
Look at satellite links to the other end of the satellite link at this station. Find trunks that are pre-emptable to that other end (from other earth stations) and open TREE entries for them.
No
Find the cost for the new segment added to the altroute by the new TREE station entries. Add to the cost to the parent station of this new entry as the cost of the altroute to that station. Estimate the cost of the unknown segment of the altroute from this station to the goal stations.
Go to the TREE file again for the lowest cost entry to expand the altroute from.
c
47

2.6.2.2 The Required Data Base--This routine makes use of the station files (SF), link files (LF), trunk files (TF) and fault files (FF). These file formats are presented in report #2 and appendix A of this report. Several changes have been made to the structure since report 2 and these are explained in appendix A. The points in the routine where these files are read and are deleted after use is clearly shown so as to allow sizing of the storage required for the running of the routine.

Figure 2-2 details the entries into file TREE which contains the altroute data from which the routine works. Each station entered into TREE has an entry like the one shown in Figure 2-2. The figure is self-explanatory. Field sizes are shown for those entries of known byte count. The costs are not sized because a choice of scale factors will be needed before the number of significant digits needed is determined. The file ALTR (used to relay the altroute) is exactly like TREE in that it contains fields 1-7 and 12 and 13 from TREE. These fields give all of the data needed to patch the altroute and create pointers to the link and trunk files showing the altroute's presence.

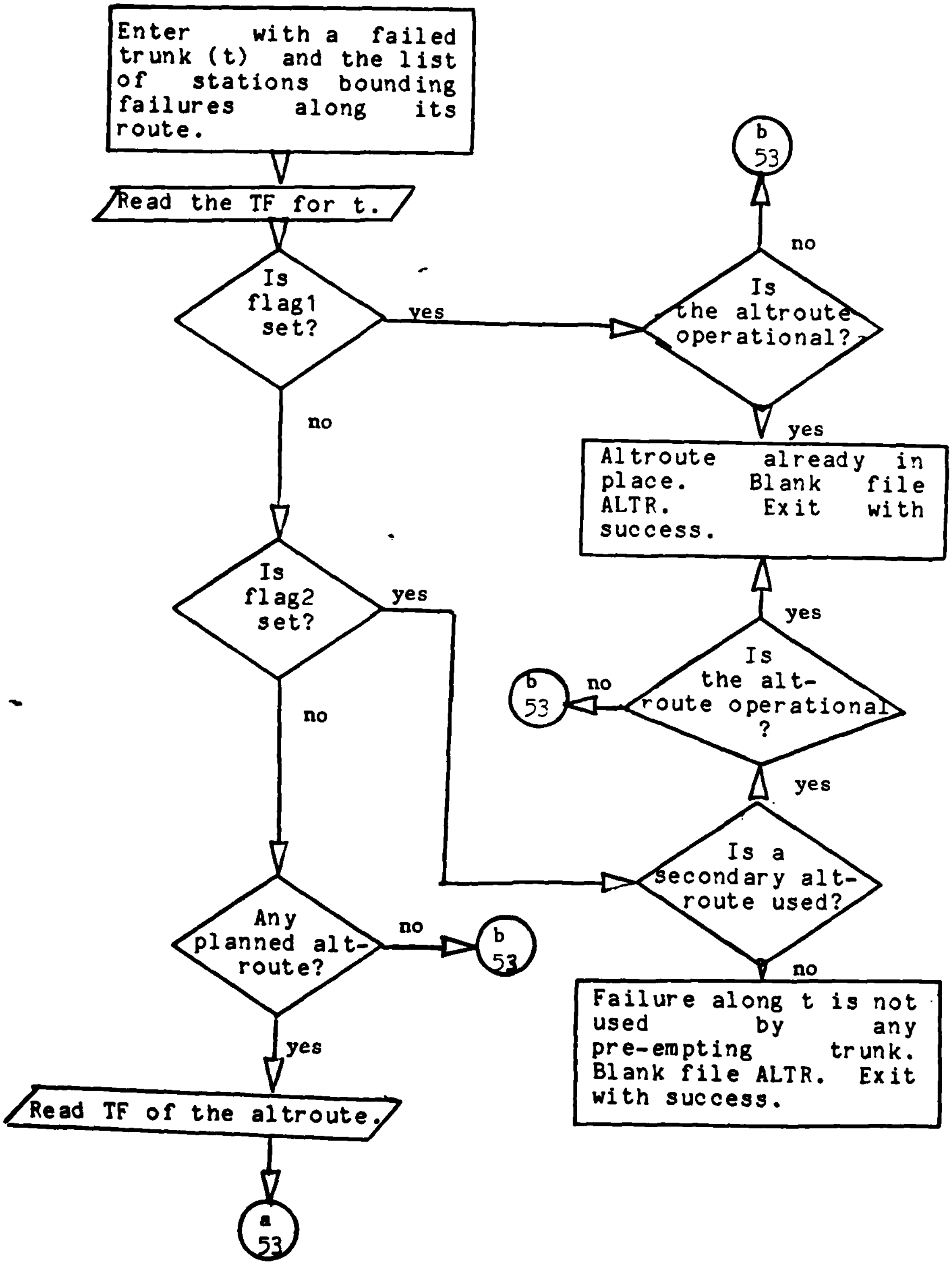
Station No.	Parent Station No.	Link no. of link leaving parent station (PS) for this station.	Group no. of group leaving PS for this station.	Link no. of link used to enter this station.	Group no. of group used to enter this station.	Trunk pre-empted to use this group.
(1) 3 bytes	(2) 3 bytes	(3) 5 bytes	(4) 1 byte	(5) 5 bytes	(6) 1 byte	(7) 8 bytes

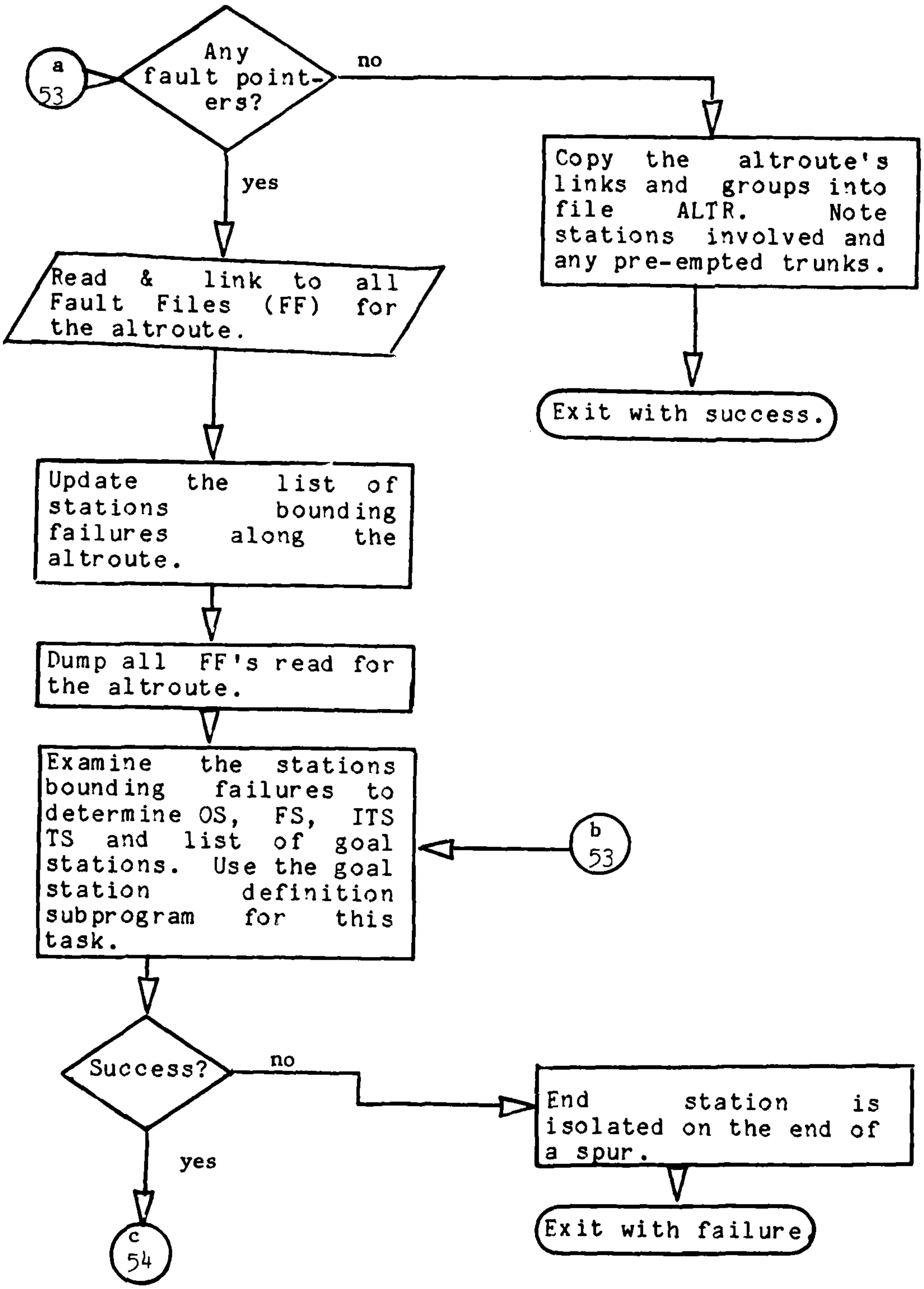
Cost to get from FS to this station via the PS. (COST)	Estimated cost to get to ITS from this station. (HITS)	Estimated cost to get to TS from this station. (HTS)	Search status	Satellite group moved as part of the link to the PS.	Satellite link carrying the group to be moved.	Connectivity path I.D. for this station (if any)	Flag indicates that a goal station is between this station and its PS on this trunk.
(8)	(9)	(10)	(11) 1 byte	(12) 8 bytes	(13) 5 bytes	(14) 2 bytes	(15) 1 byte

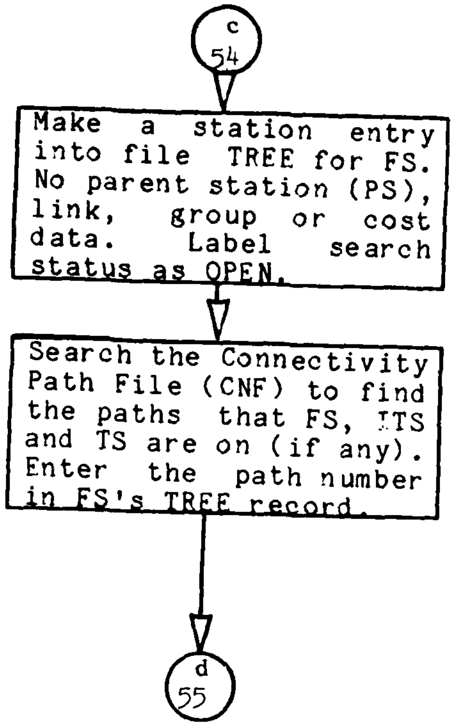
FIGURE 2-2. THE STRUCTURE OF THE ENTRIES INTO FILE "TREE"

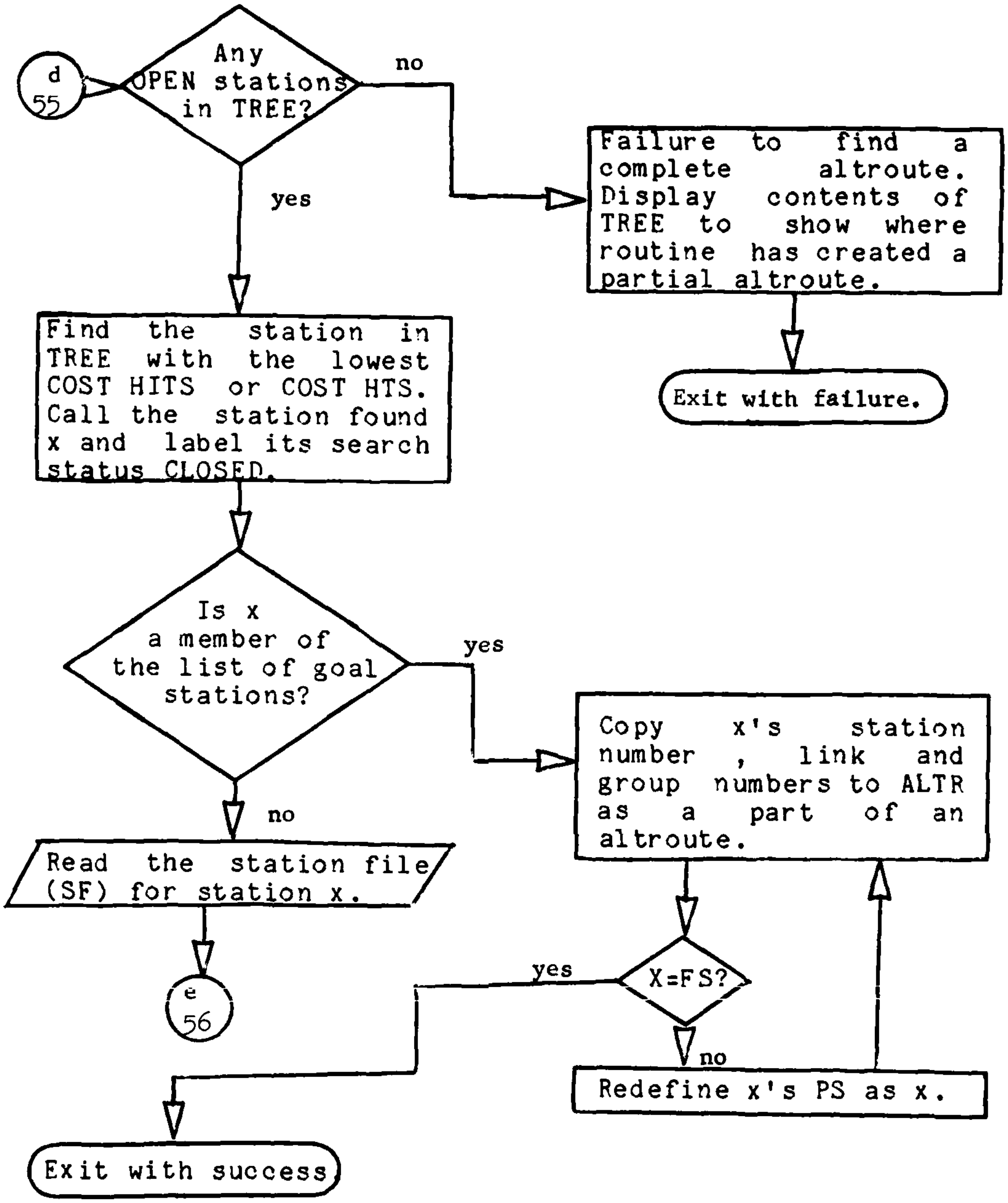
2.6.2.3

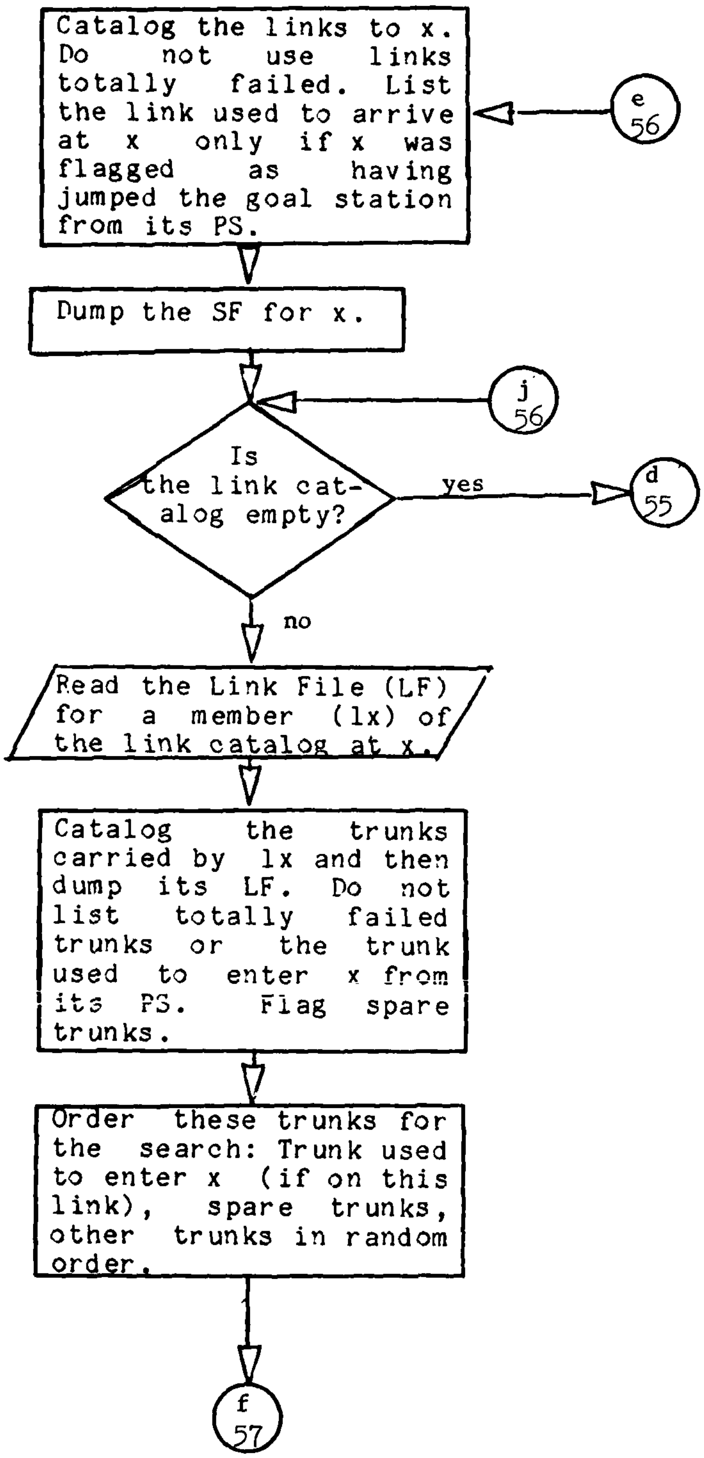
Flow Chart of the Trunk Altrouting Routine--

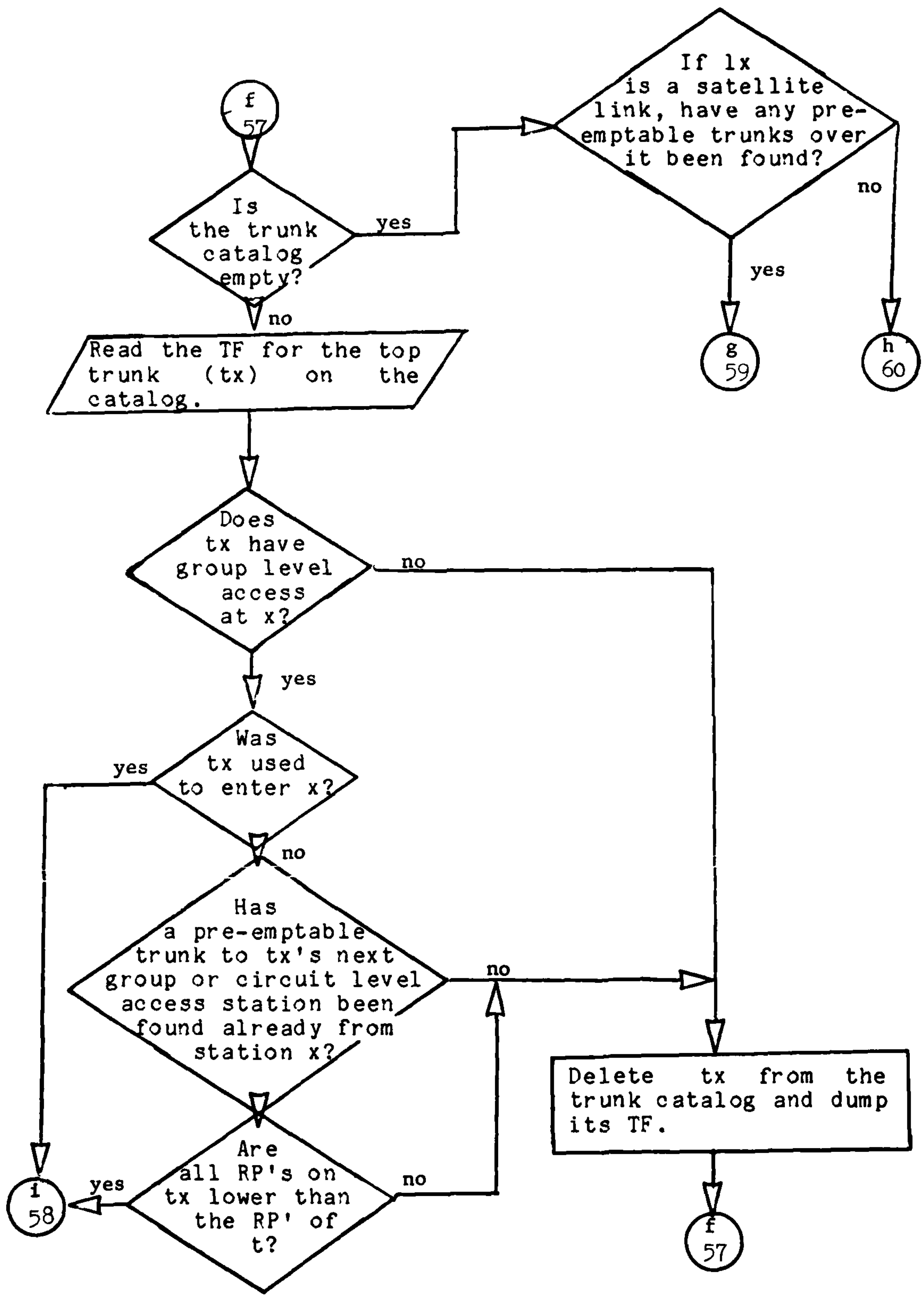


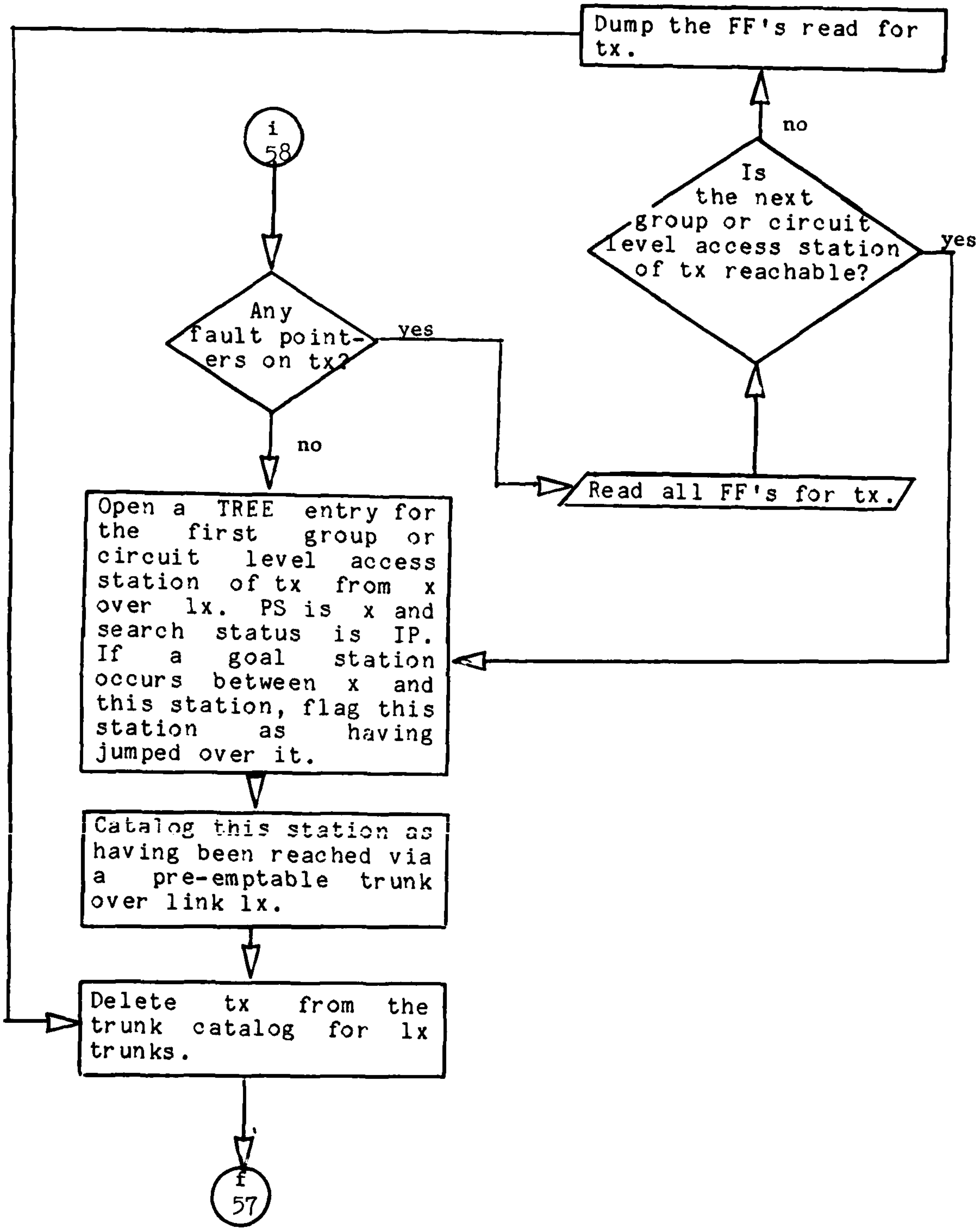

a
53
Any fault pointers?
no
yes
Copy the altroute's links and groups into file ALTR. Note stations involved and any pre-empted trunks.
Exit with success.
Read & link to all Fault Files (FF) for the altroute.
Update the list of stations bounding failures along the altroute.
Dump all FF's read for the altroute.
Examine the stations bounding failures to determine OS, FS, ITS TS and list of goal stations. Use the goal station definition subprogram for this task.
b
53
Success?
no
yes
End station is isolated on the end of a spur.
Exit with failure
c
54

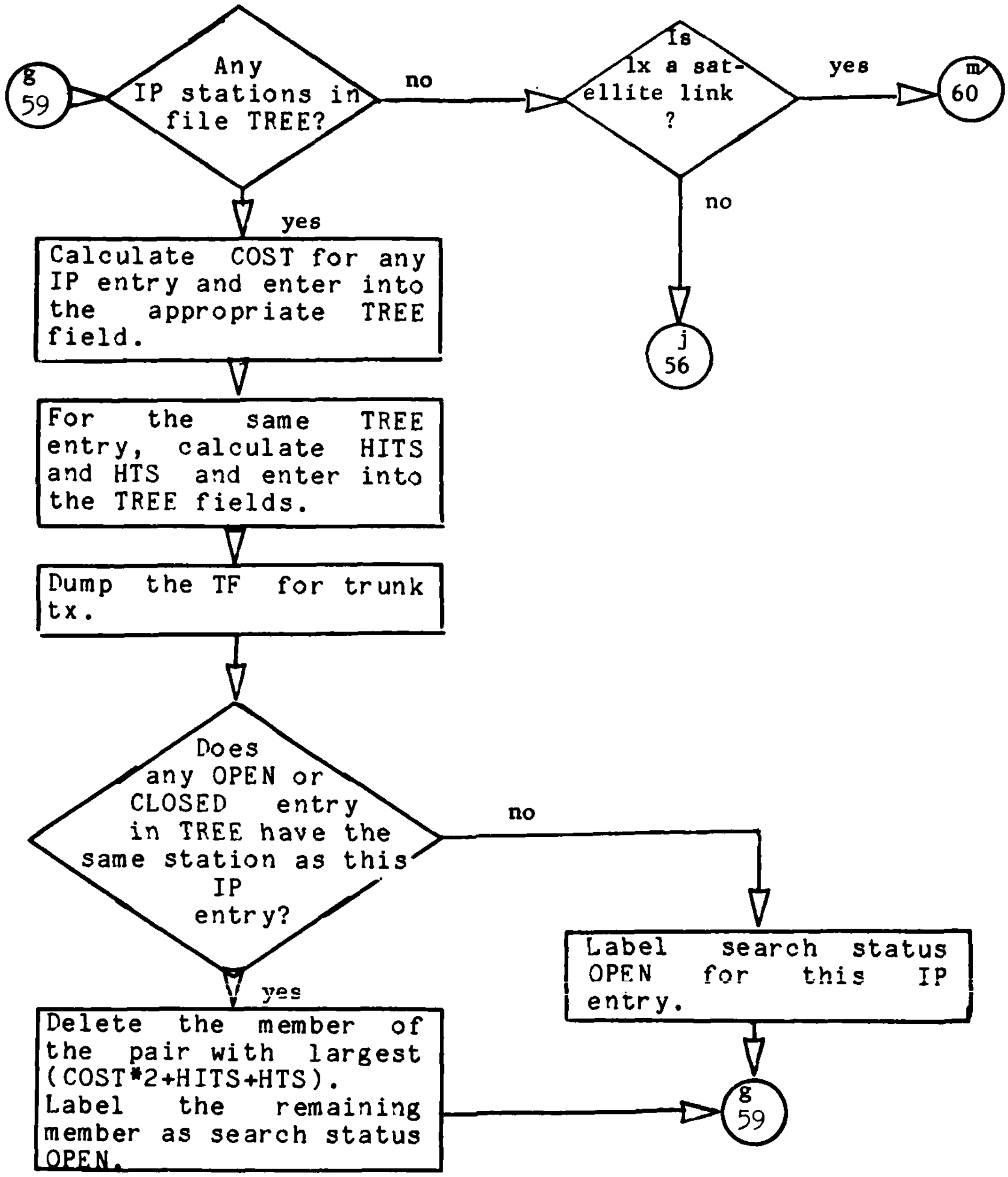

c
54
Make a station entry into file TREE for FS. No parent station (PS), link, group or cost data. Label search status as OPEN.
Search the Connectivity Path File (CNF) to find the paths that FS, ITS and TS are on (if any). Enter the path number in FS's TREE record.
d
55

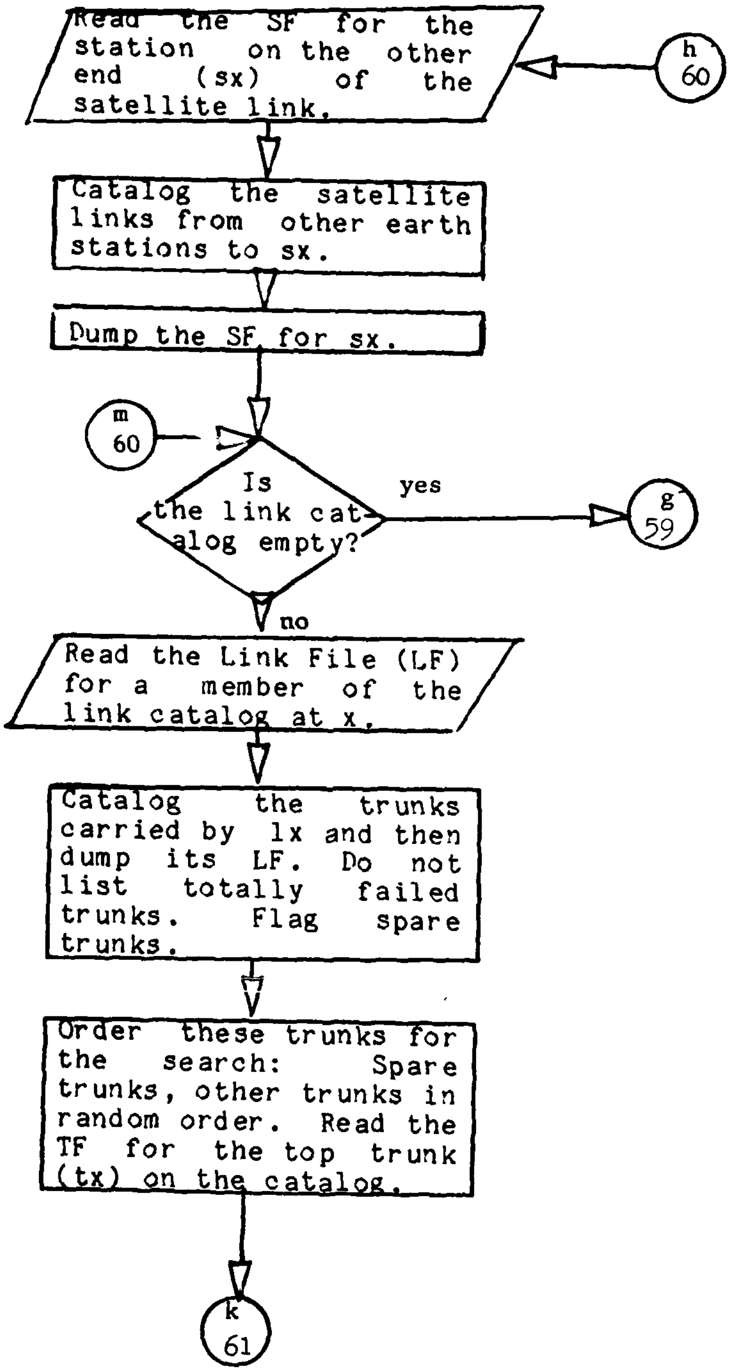

d
55
Any OPEN stations in TREE?
no
yes
Failure to find a complete altroute. Display contents of TREE to show where routine has created a partial altroute.
Exit with failure.
Find the station in TREE with the lowest COST HITS or COST HTS. Call the station found x and label its search status CLOSED.
Is x a member of the list of goal stations?
yes
no
Copy x's station number , link and group numbers to ALTR as a part of an altroute.
Read the station file (SF) for station x.
e
56
X=FS?
yes
no
Redefine x's PS as x.
Exit with success

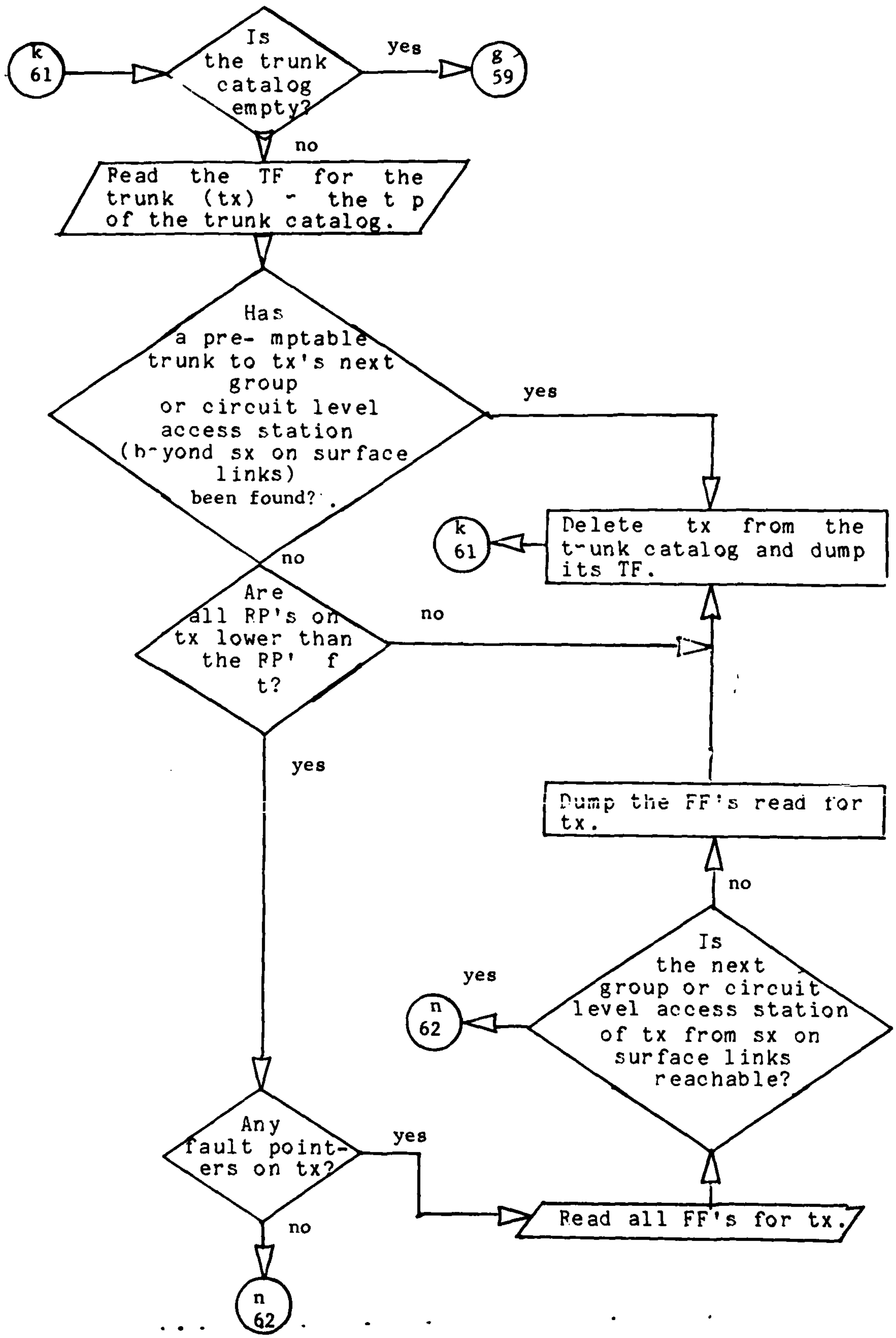

Catalog the links to x. Do not use links totally failed. List the link used to arrive at x only if x was flagged as having jumped the goal station from its PS.
e
56
Dump the SF for x.
j
56
Is the link catalog empty?
yes
d
55
no
Read the Link File (LF) for a member (lx) of the link catalog at x.
Catalog the trunks carried by lx and then dump its LF. Do not list totally failed trunks or the trunk used to enter x from its PS. Flag spare trunks.
Order these trunks for the search: Trunk used to enter x (if on this link), spare trunks, other trunks in random order.
f
57


f 57
Is the trunk catalog empty?
yes
no
If lx is a satellite link, have any pre-emptable trunks over it been found?
yes
no
g 59
h 60
Read the TF for the top trunk (tx) on the catalog.
Does tx have group level access at x?
no
yes
Was tx used to enter x?
yes
no
Has a pre-emptable trunk to tx's next group or circuit level access station been found already from station x?
no
Are all RP's on tx lower than the RP' of t?
yes
no
i 58
Delete tx from the trunk catalog and dump its TF.
f 57


Dump the FF's read for tx.
i 58
Any fault pointers on tx?
yes
no
Is the next group or circuit level access station of tx reachable?
no
yes
Read all FF's for tx.
Open a TREE entry for the first group or circuit level access station of tx from x over lx. PS is x and search status is IP. If a goal station occurs between x and this station, flag this station as having jumped over it.
Catalog this station as having been reached via a pre-emptable trunk over link lx.
Delete tx from the trunk catalog for lx trunks.
f 57


g 59
Any IP stations in file TREE?
no
Is lx a satellite link ?
yes
m 60
no
j 56
yes
Calculate COST for any IP entry and enter into the appropriate TREE field.
For the same TREE entry, calculate HITS and HTS and enter into the TREE fields.
Dump the TF for trunk tx.
Does any OPEN or CLOSED entry in TREE have the same station as this IP entry?
no
Label search status OPEN for this IP entry.
yes
Delete the member of the pair with largest (COST*2+HITS+HTS). Label the remaining member as search status OPEN.
g 59


Read the SF for the station on the other end (sx) of the satellite link.
h 60
Catalog the satellite links from other earth stations to sx.
Dump the SF for sx.
m 60
Is the link catalog empty?
yes
g 59
no
Read the Link File (LF) for a member of the link catalog at x.
Catalog the trunks carried by lx and then dump its LF. Do not list totally failed trunks. Flag spare trunks.
Order these trunks for the search: Spare trunks, other trunks in random order. Read the TF for the top trunk (tx) on the catalog.
k 61


k 61
Is the trunk catalog empty?
yes
g 59
no
Read the TF for the trunk (tx) - the t p of the trunk catalog.
Has a pre- mptable trunk to tx's next group or circuit level access station (b-yond sx on surface links) been found?
yes
Delete tx from the t-unk catalog and dump its TF.
k 61
no
Are all RP's on tx lower than the RP' f t?
no
yes
Dump the FF's read for tx.
no
Is the next group or circuit level access station of tx from sx on surface links reachable?
yes
n 62
Any fault pointers on tx?
yes
Read all FF's for tx.
no
n 62

n
62

Open a TREE entry for the first group or circuit level access station of tx from sx over surface links. PS is x and search status is IP. Enter the satellite link from x to sx as well as the surface links used to connect to the earth stations.

Catalog this station as having been reached via a pre-emptable trunk over link lx.

Delete tx from the trunk catalog for lx trunks.

k
61

2.6.3 The Circuit Altrouting Routine

2.6.3.1 Discussion of the Routine and a Simplified Flow Chart--The circuit altrouting routine is called upon whenever circuits or groups of circuits with common FS and ITS stations are required to be altrouted. This section discusses some details of the routine that will make the detailed flow chart more understandable. The inclusion of a simplified flow chart for the routine also aids in understanding the detailed routine's flow.

The routine begins by searching out pre-assigned altroutes that may exist for the circuit(s) to be altrouted. These are examined for their availability and implemented if found operational. If there is no pre-assigned altroute or the one listed is not intact, then the circuit(s) are sent into the search part of the routine to find an altroute.

The routine uses the entire goal station list found in the goal station definition routine when only one circuit is being altrouted. If a group of circuits are being altrouted together, then the goal station list is only ITS (which all have in common by virtue of their being grouped in the first place). The common FS station is the first station of any altroute, an is thus entered in TREE as the only searchable station.

The routine next enters the TREE search mode where the lowest cost altroute end station is found and examined for further expansion of the altroute. The highest RP circuit of the group is used as the guide to the lowest cost partial altroute because of DCA policy making its restoral of primary importance over all other circuits. The other members of the group will be brought along this most important circuit's altroute as far as is possible. Some may not be able to follow this circuit due to lack of pre-emptable circuits along the chosen route. However, every effort is made in the search to find as many pre-emptable circuits to every station in the altroute path expansion.

The search for stations to expand the altroute to begins with a listing of links and the trunks they carry as they leave the current station of search. The link entering the station from the partial altroute will only be considered if the current station is flagged as having jumped over a goal station from its parent station due to the trunk nature of the network. A listing of totally failed links and trunks made available from ATEC will weed out wasted efforts as well.

In finding pre-emptable circuits from the current search station, we select the lowest RP circuits on the trunks examined that have the same port types as the altroute group circuits. The matching begins with the highest RP group member and works down in RP value so that if only part of the group can be pre-empted onto a trunk, the most important circuits will be among that partial

altrouted group. If only a partial set of the altroute group can be put over a trunk, then other trunks to that station will be used to carry the remainder of the group (if another parallel trunk can be found). All trunks accessible at the current search station are examined so that every possible circuit level access station is found and that partial pre-emptable circuit groups to a station can be added by parallel trunks. The matching of circuits in the reroute group to pre-emptable circuits on a trunk is done by the matching routine given. More will be said about this matching latter on.

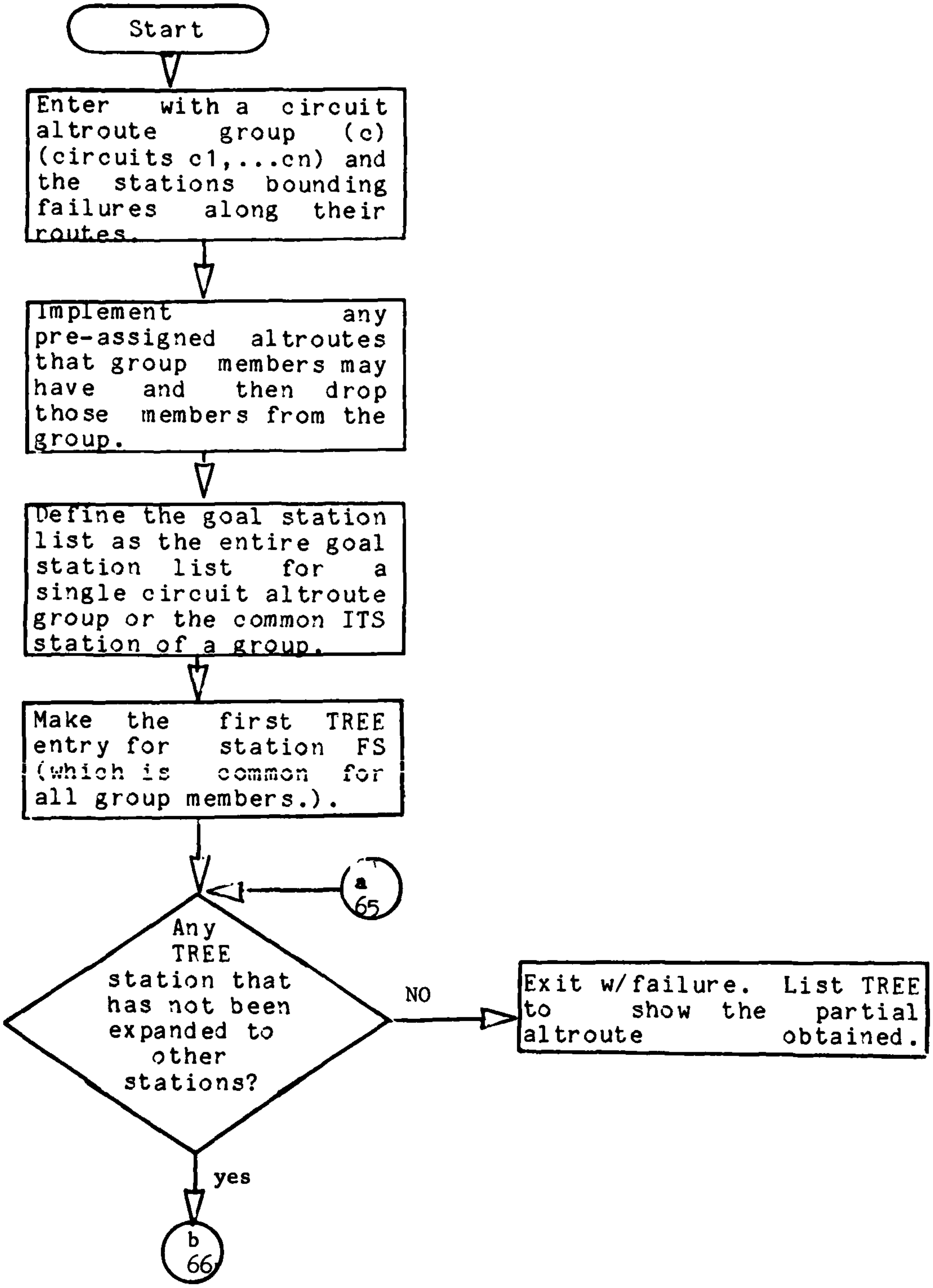
If a satellite link is accessible at the current search station and if no pre-emptable circuits were found over it, then the routine will search other satellite links to the other end of the satellite link. This is done to determine if circuits from other satellite earth stations to the other end could be pre-empted and have their channels moved to the current search station's earth terminal.

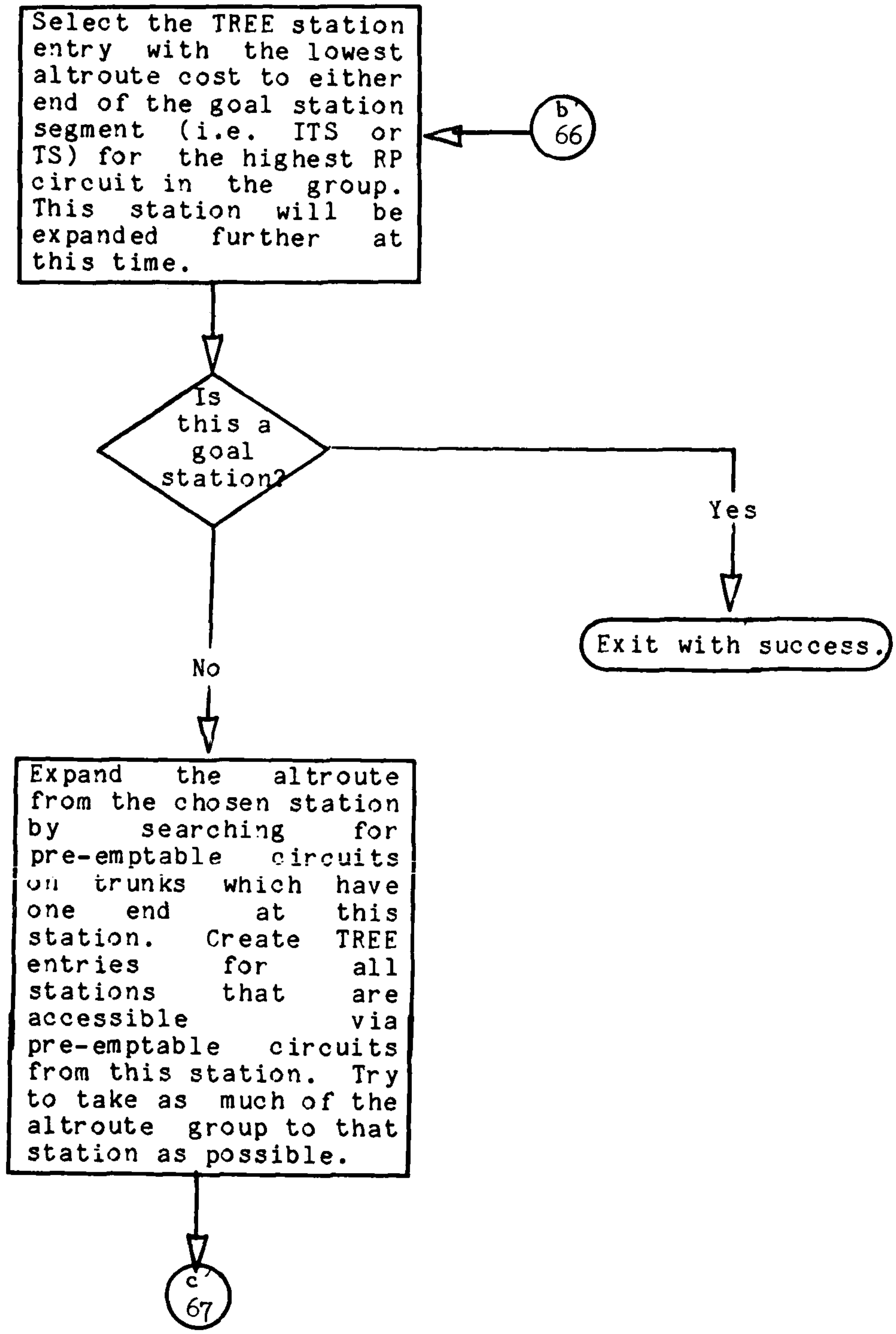
Once the station's path expansions are entered into TREE, the costs of the altroute to those stations is calculated and added to the current station's cost to give the known altroute's segment cost. The estimates for the unknown segment from each expanded station is also computed and entered into TREE. The routine then returns to TREE to find the lowest cost altroute station to expand the altroute.

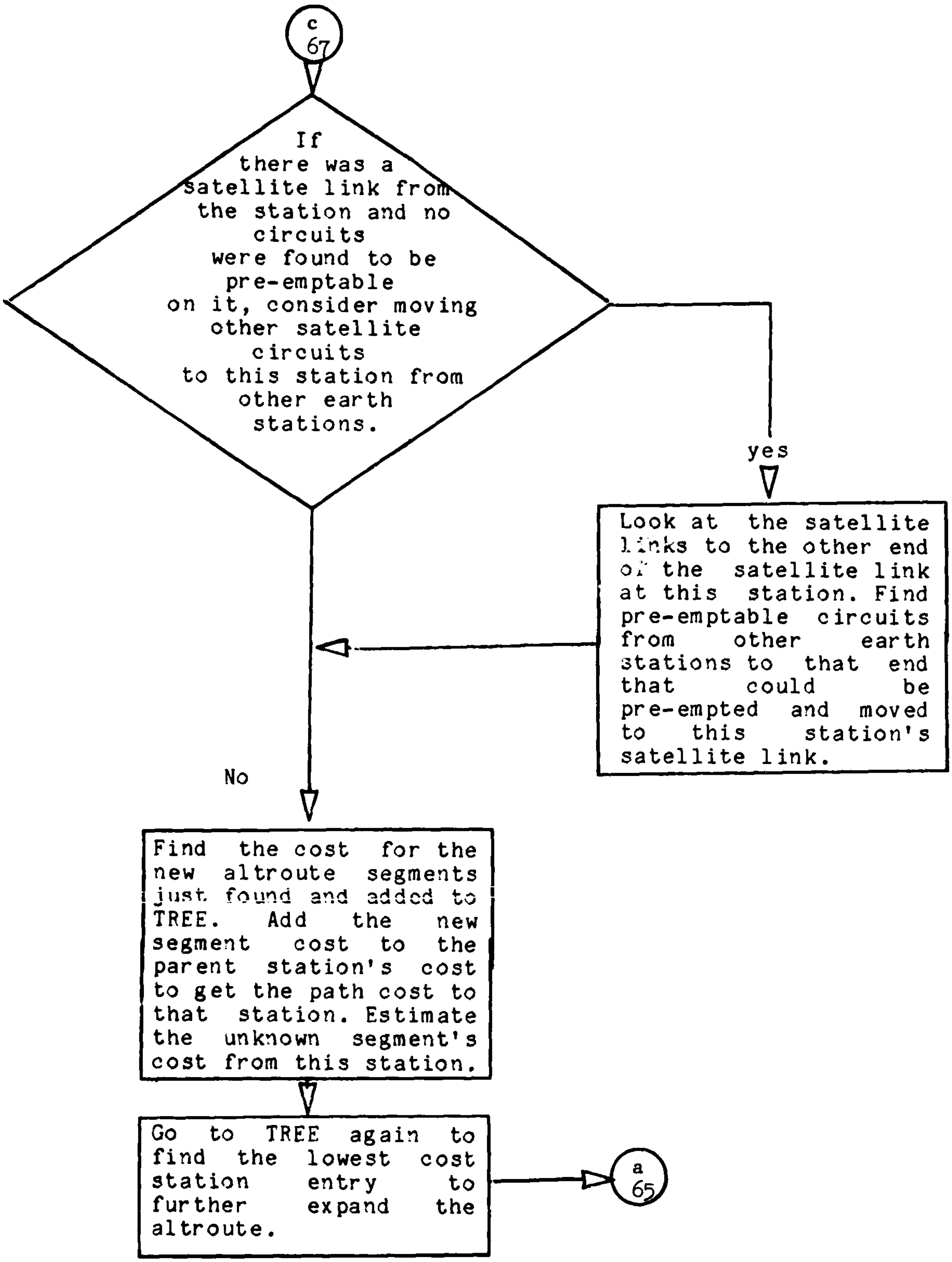
Once a station on the goal list is found as the lowest cost station in TREE, the altroute of minimum cost is found. Tracing back via the parent station pointer in each station's TREE entry reveals the altroute path. If no station exists in TREE that has not been expanded and the goal stations are still not encountered, then failure to altroute is the result. The TREE contents will provide at least a partial altroute path which may aid operations personnel in changing some RP's to allow a full altroute or some other override activity to force a full altroute.

Finally a few comments about the circuit matching subroutine attached to this altrouting routine. This routine selects all of the circuits of the altroute group of the same port type. It selects an equal number of circuits from the bottom of the RP list of the pre-empt trunk circuit list. Matching starts with the highest RP circuit in the altroute group. The lowest RP circuit in the pre-empt list that that circuit can pre-empt is matched to the circuit. The process proceeds down the altroute list in RP order. The same is done for the other port types in the altroute group. This method guarantees that the most important circuits will have first chance at a pre-emptable circuit if the number available does not match the number in the altroute group.

Simpified Flow Chart of the Circuit Altrouting Routine




Select the TREE station entry with the lowest altroute cost to either end of the goal station segment (i.e. ITS or TS) for the highest RP circuit in the group. This station will be expanded further at this time.
b
66
Is this a goal station?
Yes
Exit with success.
No
Expand the altroute from the chosen station by searching for pre-emptable circuits on trunks which have one end at this station. Create TREE entries for all stations that are accessible via pre-emptable circuits from this station. Try to take as much of the altroute group to that station as possible.
c
67


c 67
If there was a satellite link from the station and no circuits were found to be pre-emptable on it, consider moving other satellite circuits to this station from other earth stations.
yes
Look at the satellite links to the other end of the satellite link at this station. Find pre-emptable circuits from other earth stations to that end that could be pre-empted and moved to this station's satellite link.
No
Find the cost for the new altroute segments just found and added to TREE. Add the new segment cost to the parent station's cost to get the path cost to that station. Estimate the unknown segment's cost from this station.
Go to TREE again to find the lowest cost station entry to further expand the altroute.
a 65

2.6.3.2 <u>The Required Data Base</u>--This routine makes use of station, link, trunk, and fault files as described in Appendix A. In addition, a number of catalogs of items is kept. The most complex of these is the TREF file and the pre-emptable circuit list file. These will be discussed in detail because of their complexity. The other lists are simple and easily understood in the context of the flow charts.

The TREF file detail is shown in Figure 2-3. This file is similar to the TREE file used for the trunk altrouting routine. Each station entry has a set of fields for each circuit of the altroute group. There are a few fields common to all circuits in the group and these are separate from the individual circuit's fields. The subset of fields 1-10, 15, 16, and 18 are placed into file ALTR as the altroute output. These fields provide all of the data necessary to patch the altroute and link the altroutes to the appropriate trunk and circuit files in the data base to make the altroutes visible.

The file of pre-emptable circuit lists has the detailed description in Figure 2-4. There is one such entry for every station accessible at circuit level from the current search station. If more than one trunk is necessary to provide pre-emptable circuits to a station, then there will be one such entry for each of the parallel trunks to a station from the current search station.

Station no.	Parent station no.	Link no. of link leaving parent station (PS) for this station.	Group no. of group leaving PS for this station	Channel no. of channel leaving PS for this station	Link no. of link used to enter this station.	Group no. of group used to enter this station	Channel no. of channel used to enter this station.	Circuit pre-empted to use this channel.	Trunk carrying pre-empted circuit.
(1) 3 bytes	(2) 3 bytes	(3) 5 bytes	(4) 1 byte	(5) 2 bytes	(6) 5 bytes	(7) 1 byte	(8) 2 bytes	(9) 4 bytes	(10) 8 bytes

Cost to get from FS to this station via the PS. (COST)	Estimated cost to get to ITS from this station. (HITS)	Estimated cost to get to TS from this station. (HTS)	Search status	Satellite channel moved as part of the link to the PS.	Satellite group the moved channel occupies	. . . (Data like the above fields for the other circuits in the re-route group being handled. Enter in the order of the circuit's RP's.)
(11)	(12)	(13)	(14) 1 byte	(15) 2 bytes	(16) 1 byte	(17)

FIGURE 2-3. THE STRUCTURE OF THE ENTRIES INTO FILE "TREE" FOR CIRCUIT ALTROUTING

. . . (Data for last circuit in the altroute list.)		Satellite link the moved circuits occupy.	Connectivity path I.D. 5 for this station (if any). (up to 5 listed)	Flag indicating that a goal station is between this station and its PS on this trunk
		(18) 5 bytes	(19) 10 bytes	(20) 1 byte

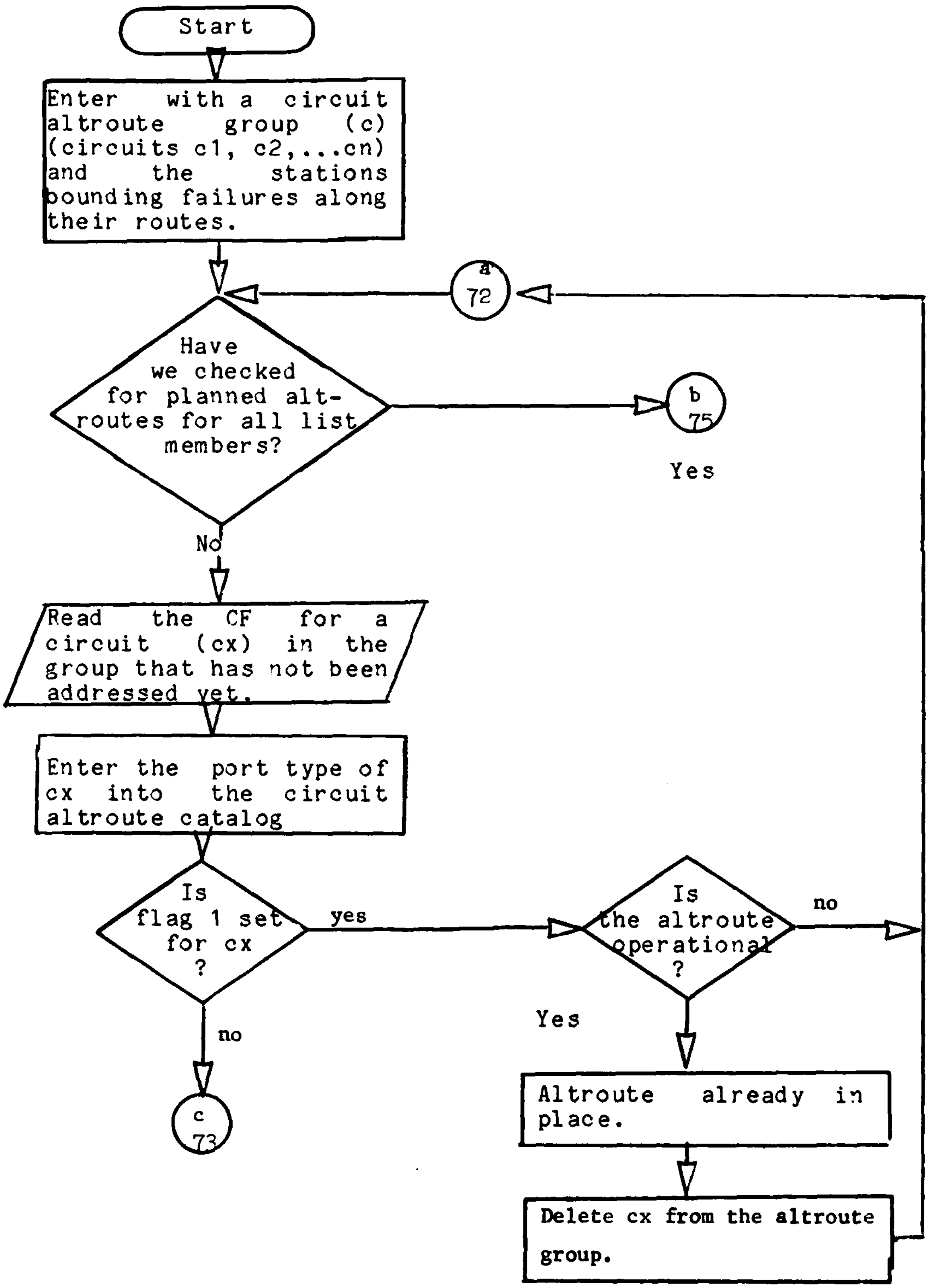
Figure 2-3 (continued)

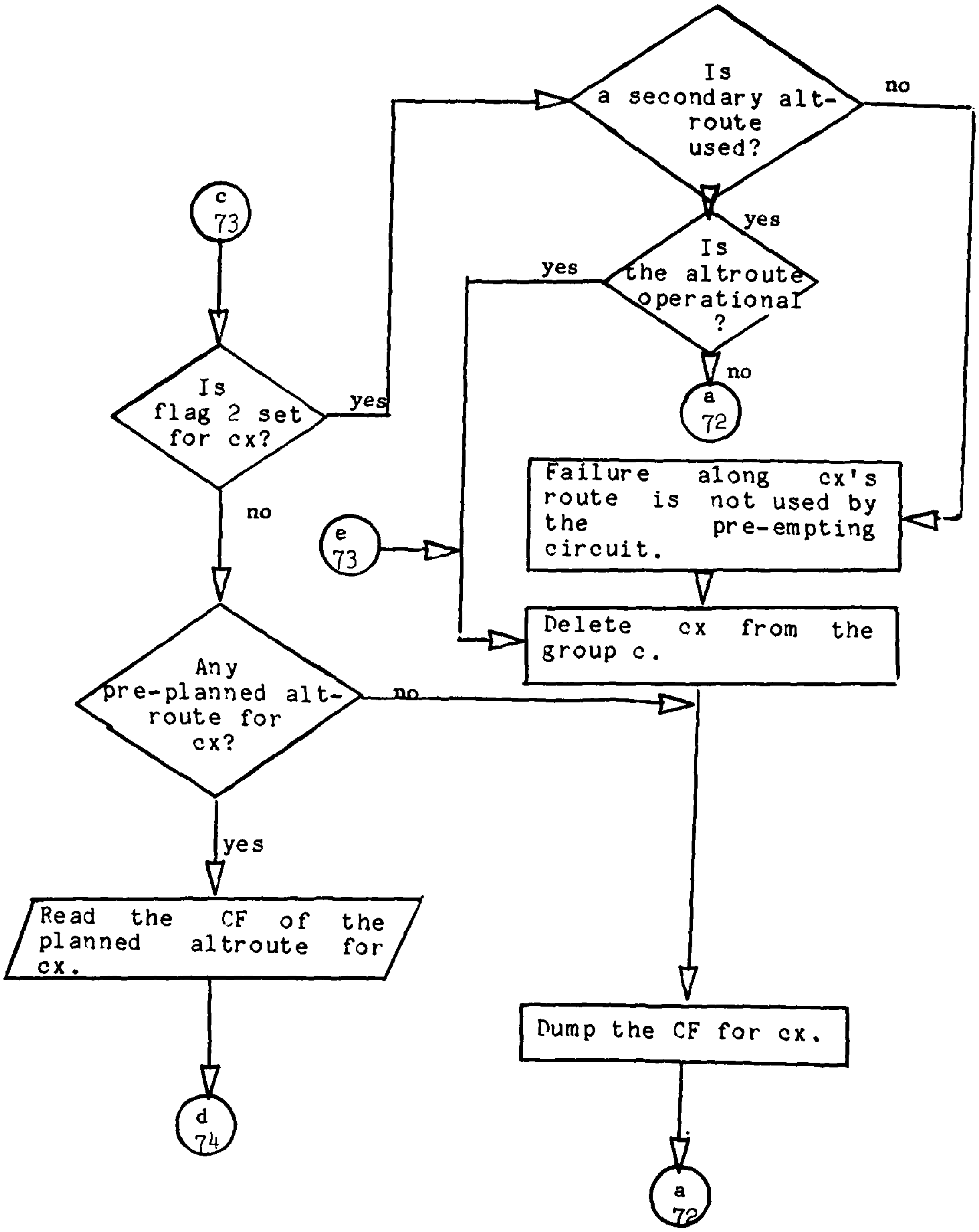
(One Entry as Shown Below for Every Trunk to Every Station Accessible from the Current Search Station.)

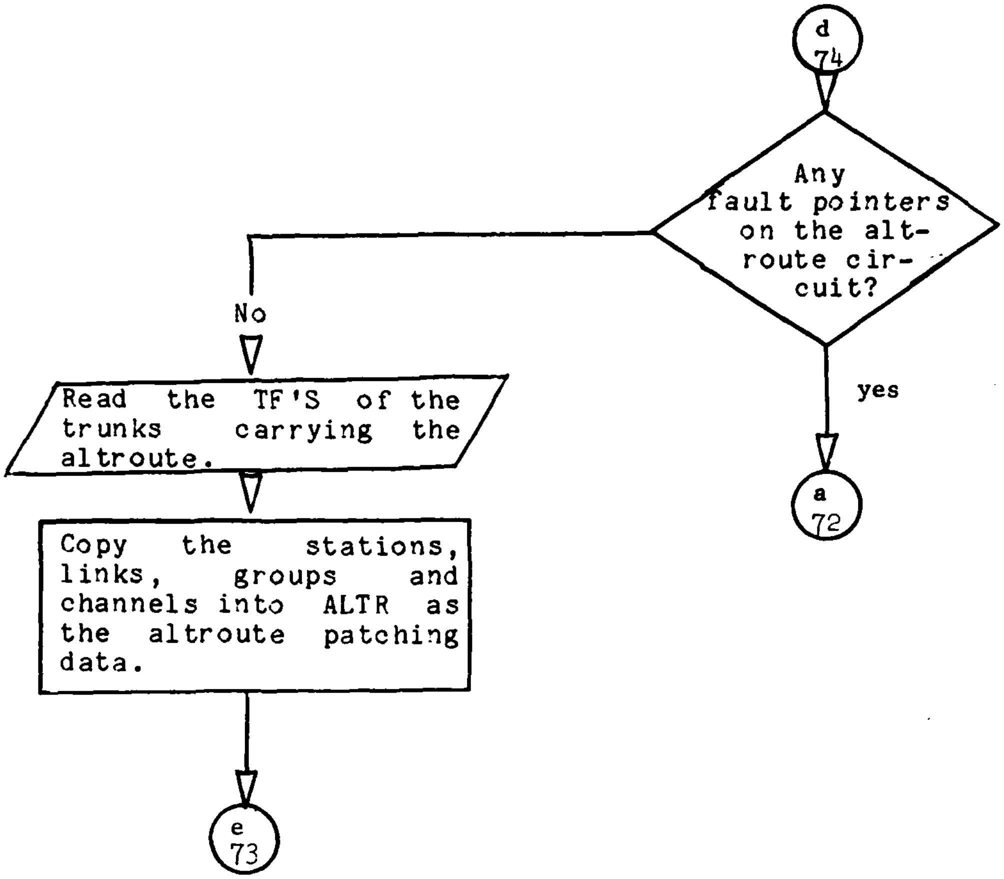
Station accessible over the listed pre-pre-emptable circuits.	Trunk carrying the listed pre-emptable circuits.	CCSD of pre-emptable circuit 4 bytes	CCSD of altroute circuit that pre-empts. 4 bytes	List of circuits pre-emptable to the same station from the current search station. (24 max)	. . .	(Next station's entry)
(1) 3 bytes	(2) 8 bytes	(3) 192 bytes				

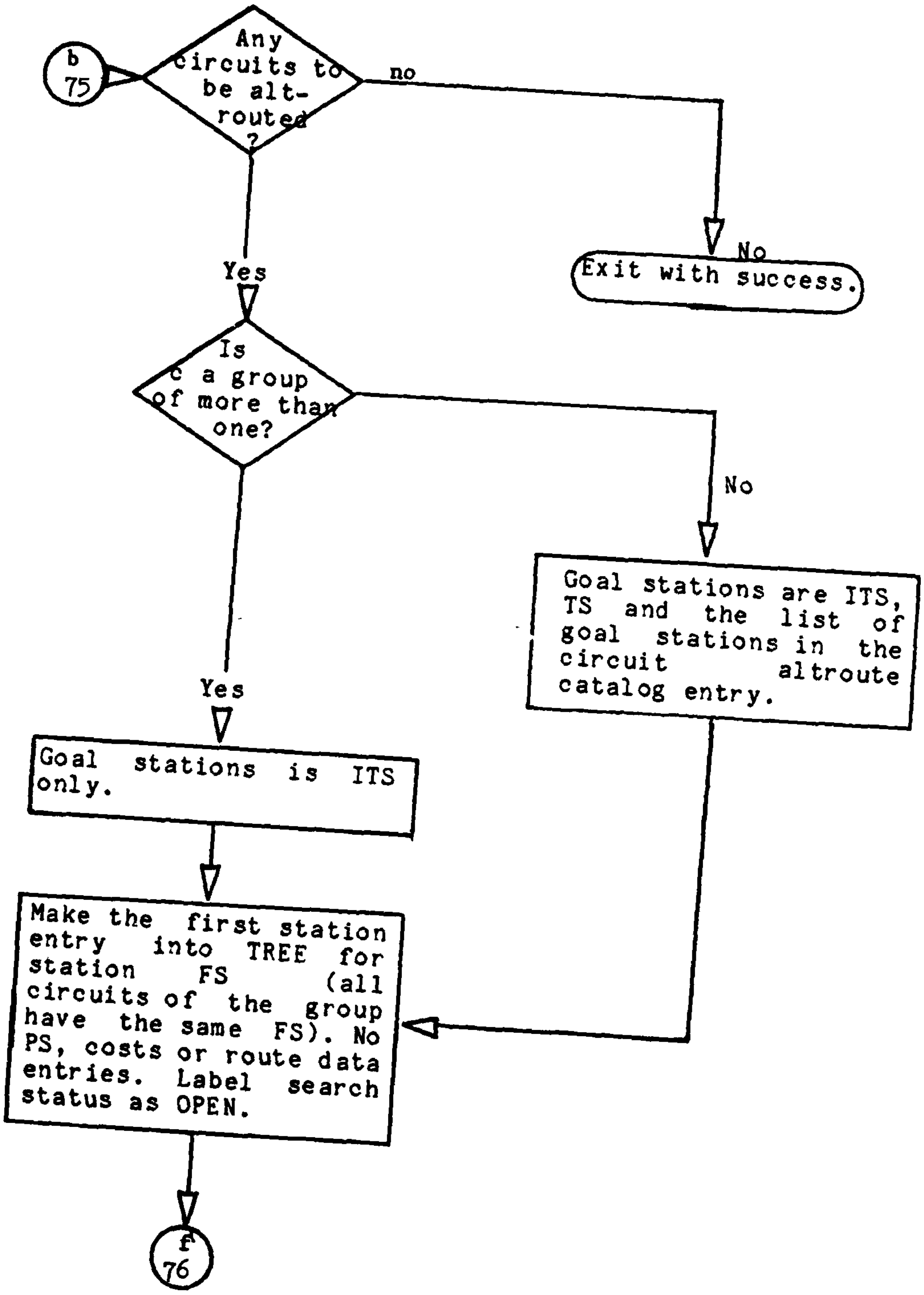
FIGURE 2-4. STRUCTURE OF THE PRE-EMPTABLE CIRCUIT LIST FILE

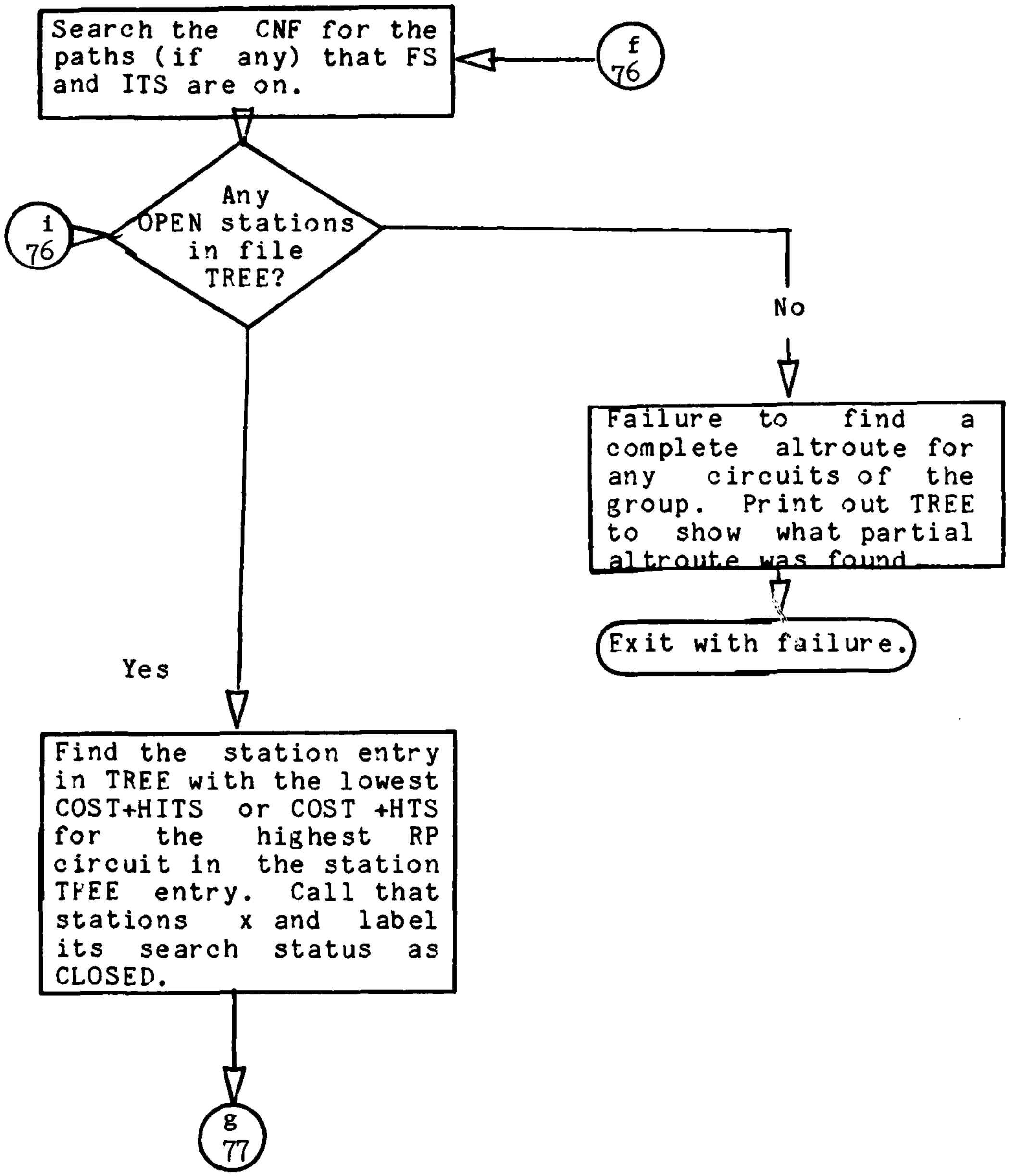
2.6.3.3
The Detailed Flow Chart of the Circuit Altrouting Routine

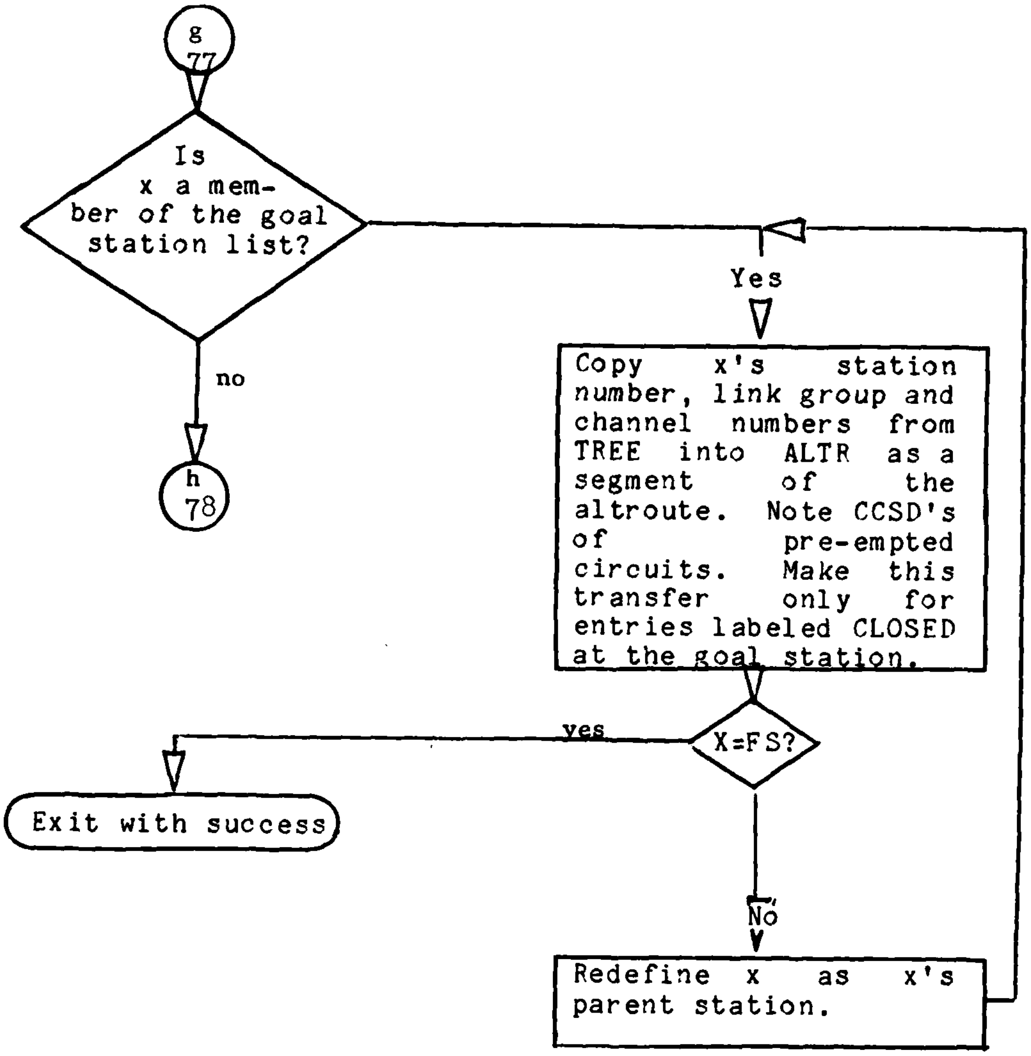


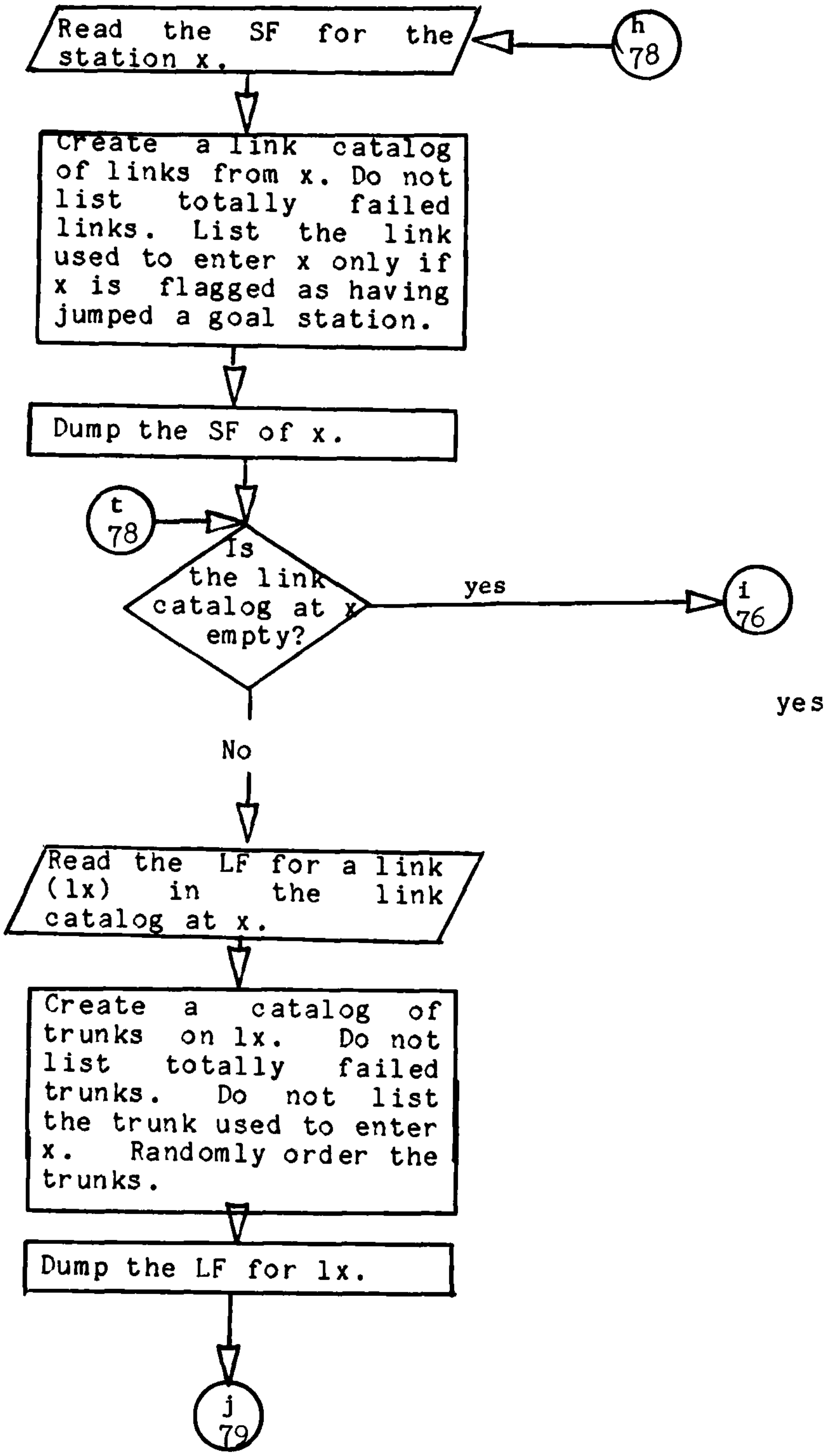

Is a secondary alt-route used?
no
yes
c 73
Is the altroute operational ?
yes
no
a 72
Is flag 2 set for cx?
yes
no
e 73
Failure along cx's route is not used by the pre-empting circuit.
Delete cx from the group c.
Any pre-planned alt-route for cx?
no
yes
Read the CF of the planned altroute for cx.
Dump the CF for cx.
d 74
a 72

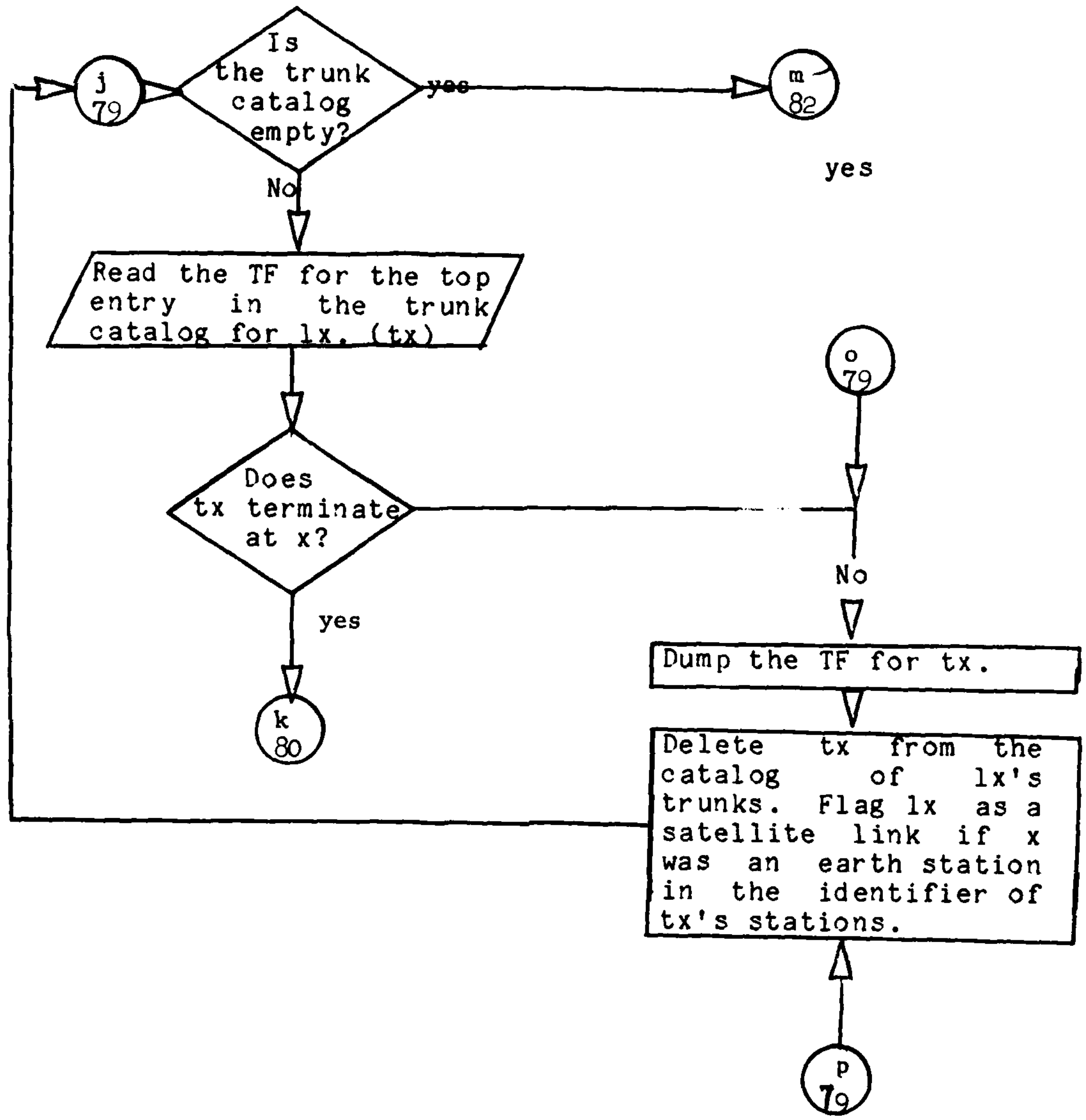

d 74
Any fault pointers on the alt- route cir- cuit?
No
yes
a 72
Read the TF'S of the trunks carrying the altroute.
Copy the stations, links, groups and channels into ALTR as the altroute patching data.
e 73


b 75
Any circuits to be alt-routed ?
no
No
Exit with success.
Yes
Is c a group of more than one?
No
Goal stations are ITS, TS and the list of goal stations in the circuit altroute catalog entry.
Yes
Goal stations is ITS only.
Make the first station entry into TREE for station FS (all circuits of the group have the same FS). No PS, costs or route data entries. Label search status as OPEN.
f 76

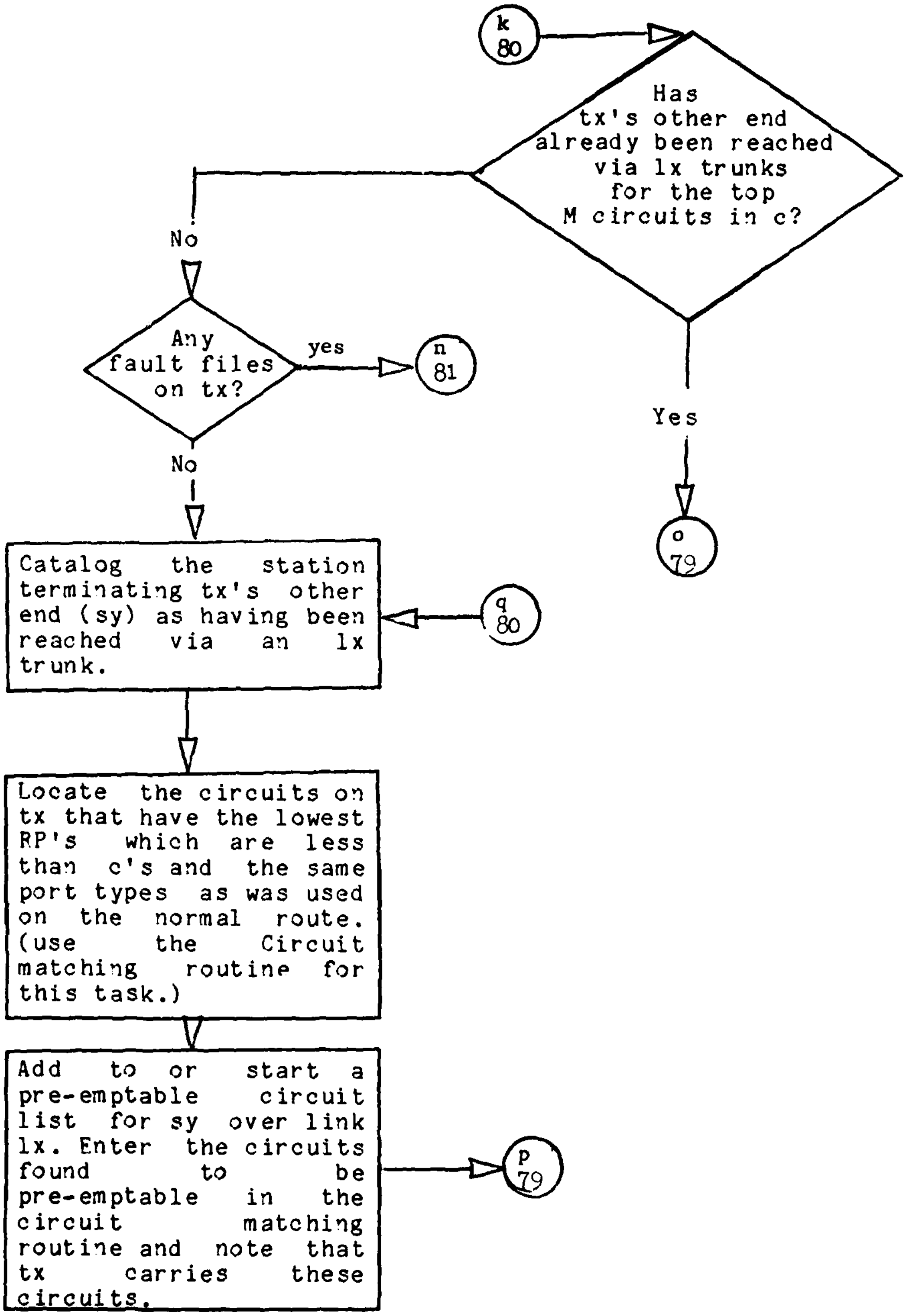

Search the CNF for the paths (if any) that FS and ITS are on.
f 76
i 76
Any OPEN stations in file TREE?
No
Failure to find a complete altroute for any circuits of the group. Print out TREE to show what partial altroute was found.
Exit with failure.
Yes
Find the station entry in TREE with the lowest COST+HITS or COST +HTS for the highest RP circuit in the station TREE entry. Call that stations x and label its search status as CLOSED.
g 77

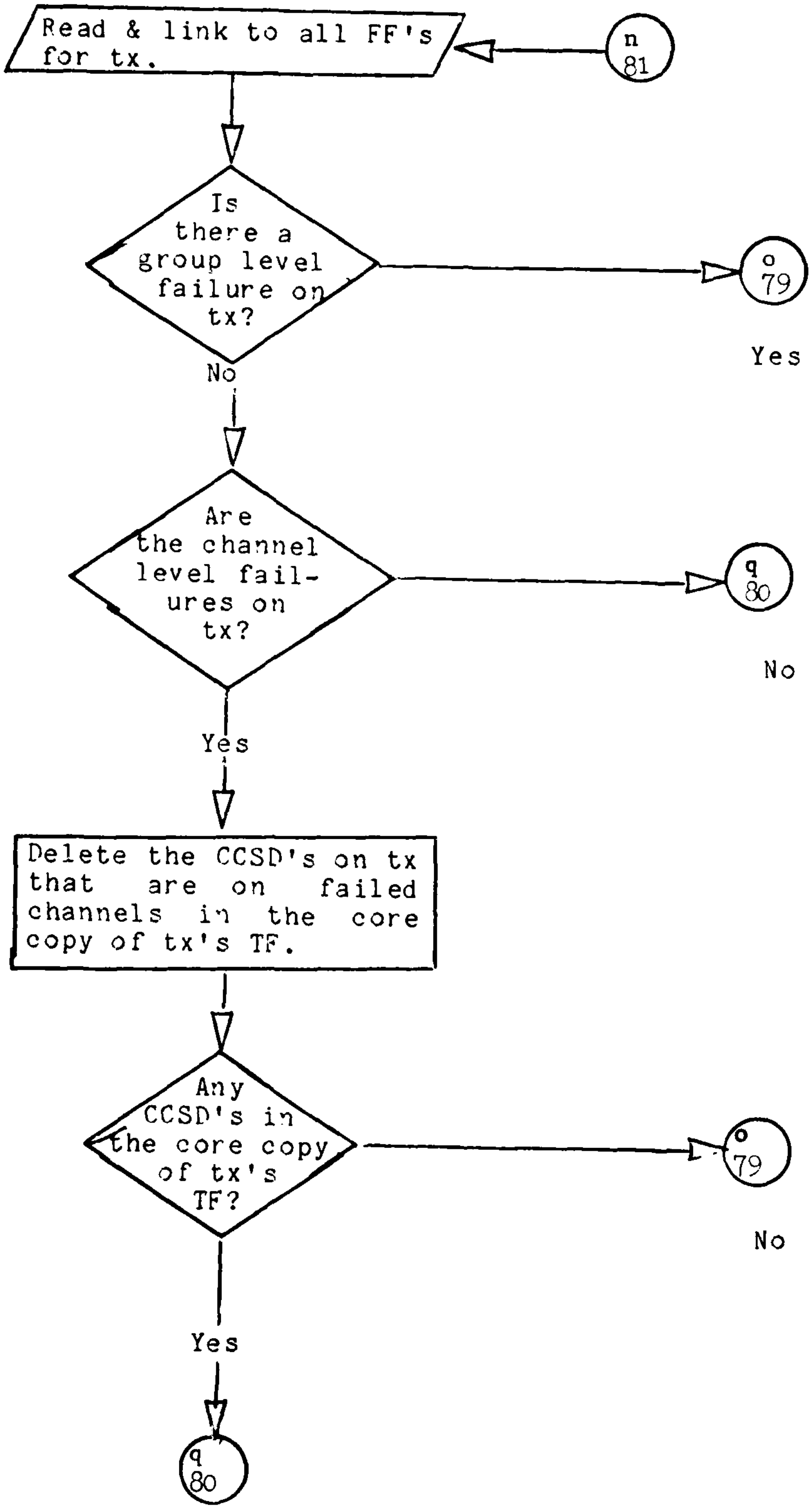

g 77
Is x a member of the goal station list?
Yes
no
h 78
Copy x's station number, link group and channel numbers from TREE into ALTR as a segment of the altroute. Note CCSD's of pre-empted circuits. Make this transfer only for entries labeled CLOSED at the goal station.
X=FS?
yes
Exit with success
No
Redefine x as x's parent station.


Read the SF for the station x.
h 78
Create a link catalog of links from x. Do not list totally failed links. List the link used to enter x only if x is flagged as having jumped a goal station.
Dump the SF of x.
t 78
Is the link catalog at x empty?
yes
i 76
yes
No
Read the LF for a link (lx) in the link catalog at x.
Create a catalog of trunks on lx. Do not list totally failed trunks. Do not list the trunk used to enter x. Randomly order the trunks.
Dump the LF for lx.
j 79



j 79
Is the trunk catalog empty?
yes
m 82
yes
No
Read the TF for the top entry in the trunk catalog for lx. (tx)
o 79
Does tx terminate at x?
No
yes
Dump the TF for tx.
k 80
Delete tx from the catalog of lx's trunks. Flag lx as a satellite link if x was an earth station in the identifier of tx's stations.
p 79


k 80
Has tx's other end already been reached via lx trunks for the top M circuits in c?
No
Yes
o 79
Any fault files on tx?
yes
n 81
No
Catalog the station terminating tx's other end (sy) as having been reached via an lx trunk.
q 80
Locate the circuits on tx that have the lowest RP's which are less than c's and the same port types as was used on the normal route. (use the Circuit matching routine for this task.)
Add to or start a pre-emptable circuit list for sy over link lx. Enter the circuits found to be pre-emptable in the circuit matching routine and note that tx carries these circuits.
p 79

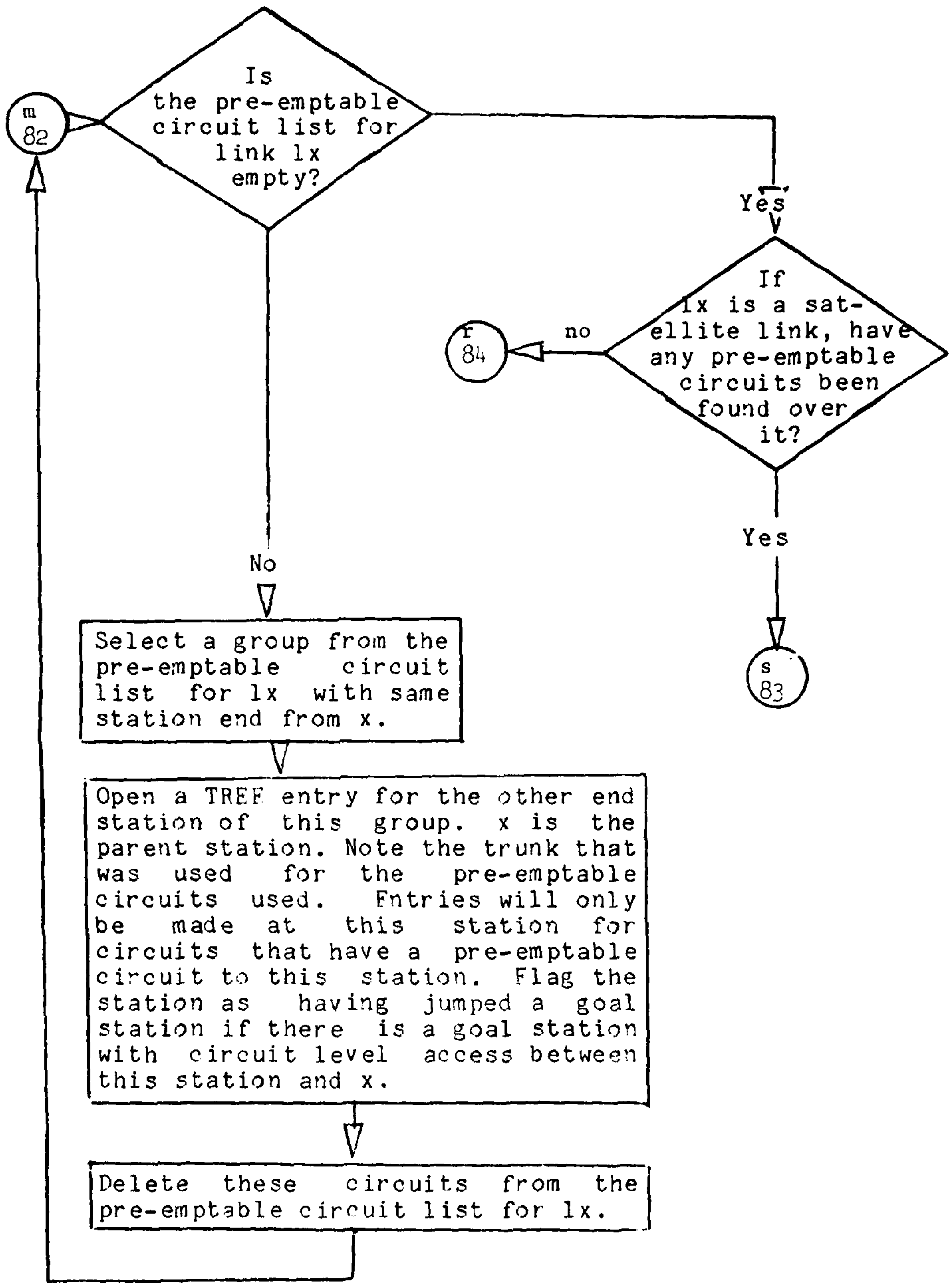

Read & link to all FF's for tx.
n
81
Is there a group level failure on tx?
o
79
Yes
No
Are the channel level failures on tx?
q
80
No
Yes
Delete the CCSD's on tx that are on failed channels in the core copy of tx's TF.
Any CCSD's in the core copy of tx's TF?
o
79
No
Yes
q
80

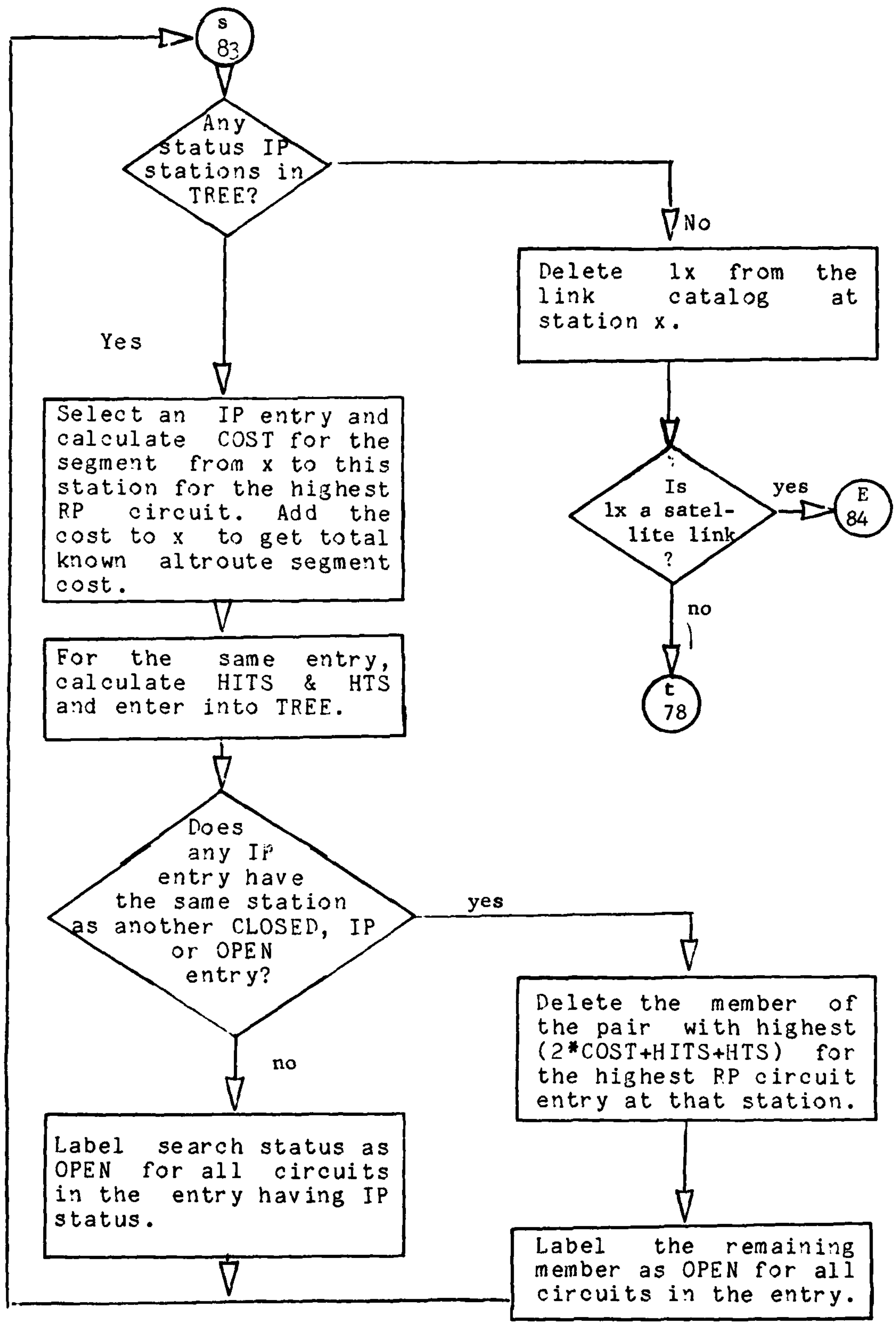
AD-A080 767 HONEYWELL SYSTEMS AND RESEARCH CENTER MINNEAPOLIS MN F/G 17/2
SYSTEM CONTROL FOR THE TRANSITIONAL DCS.(U)
MAY 79 F C ANNAND, D E DOTY, R V KRZYZANOWSKI DCA100-78-C-0017

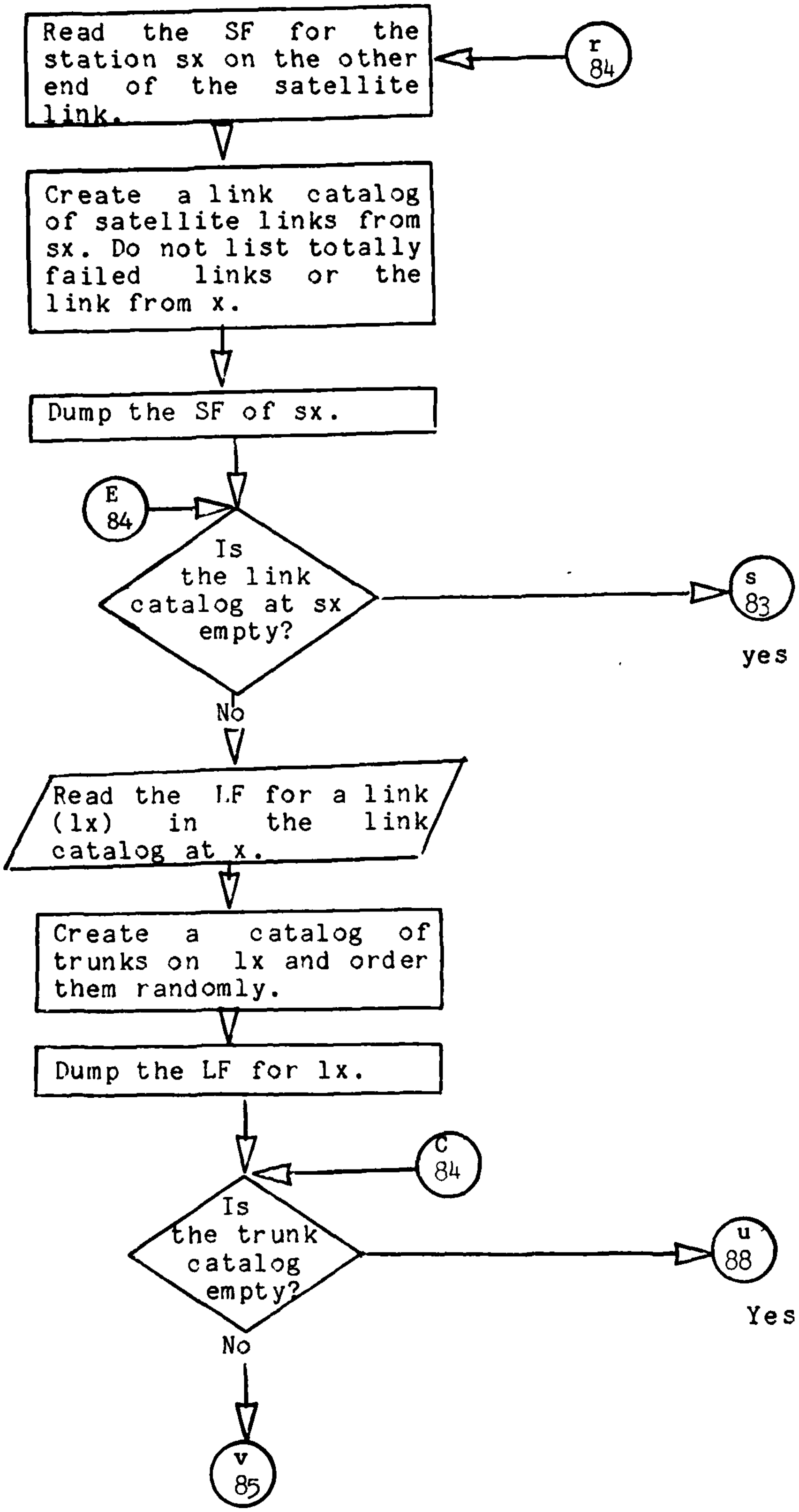
UNCLASSIFIED TR-3 SBIE-AD-E100 328 NL

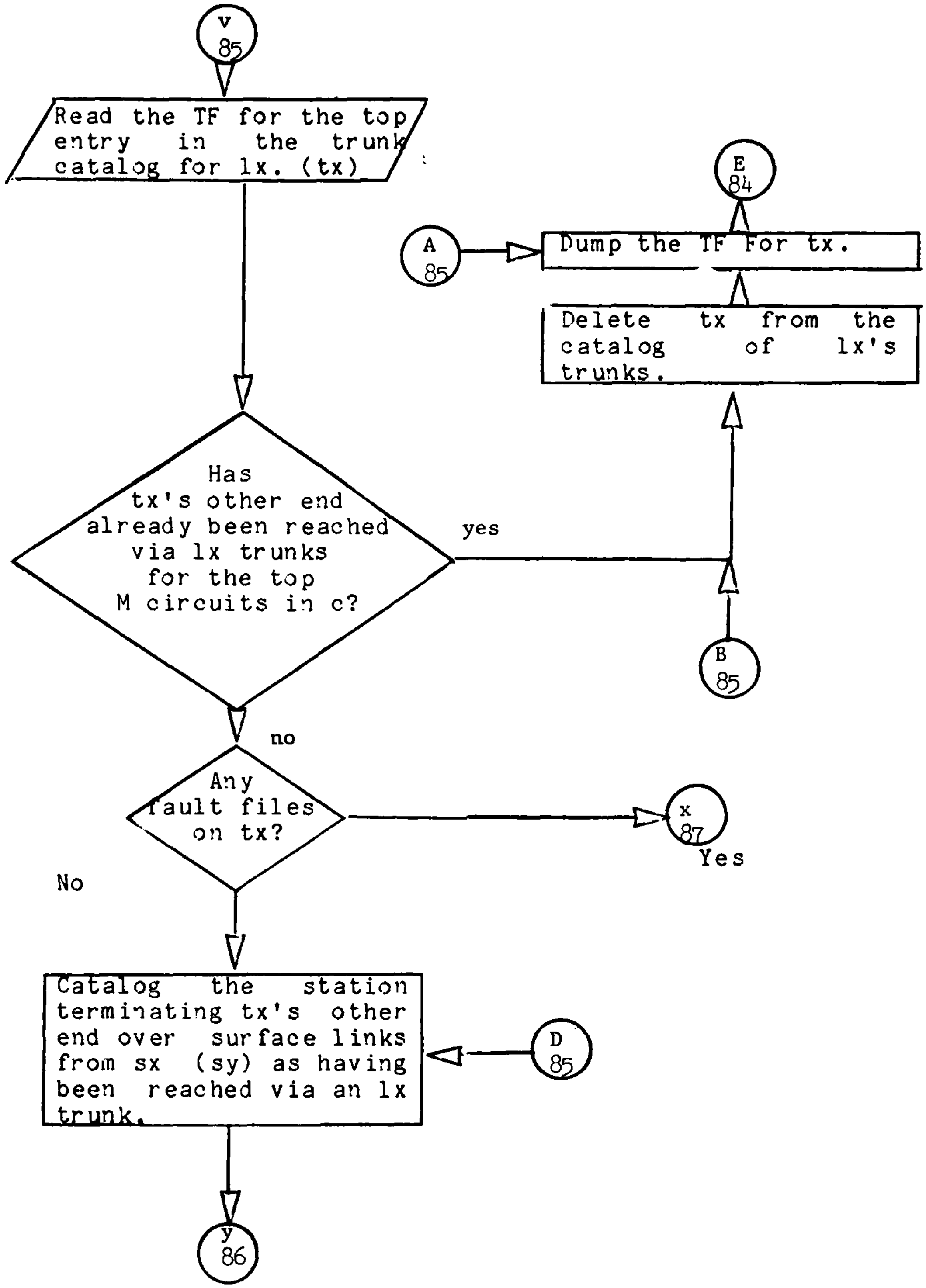
2 OF 4

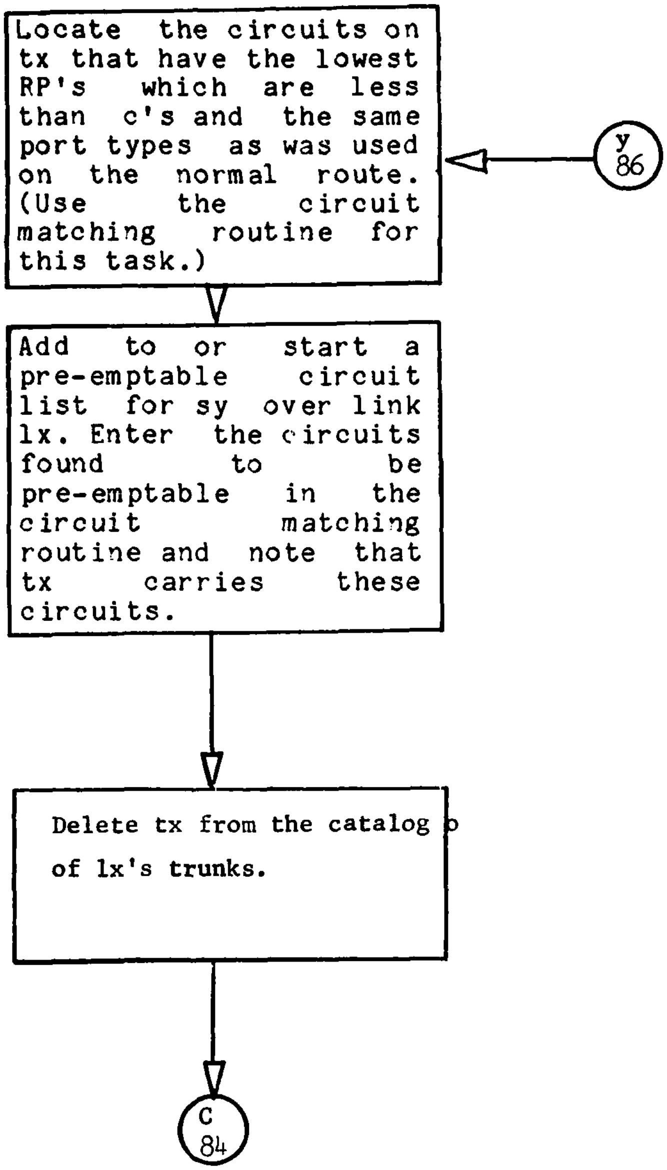
AD
A080767

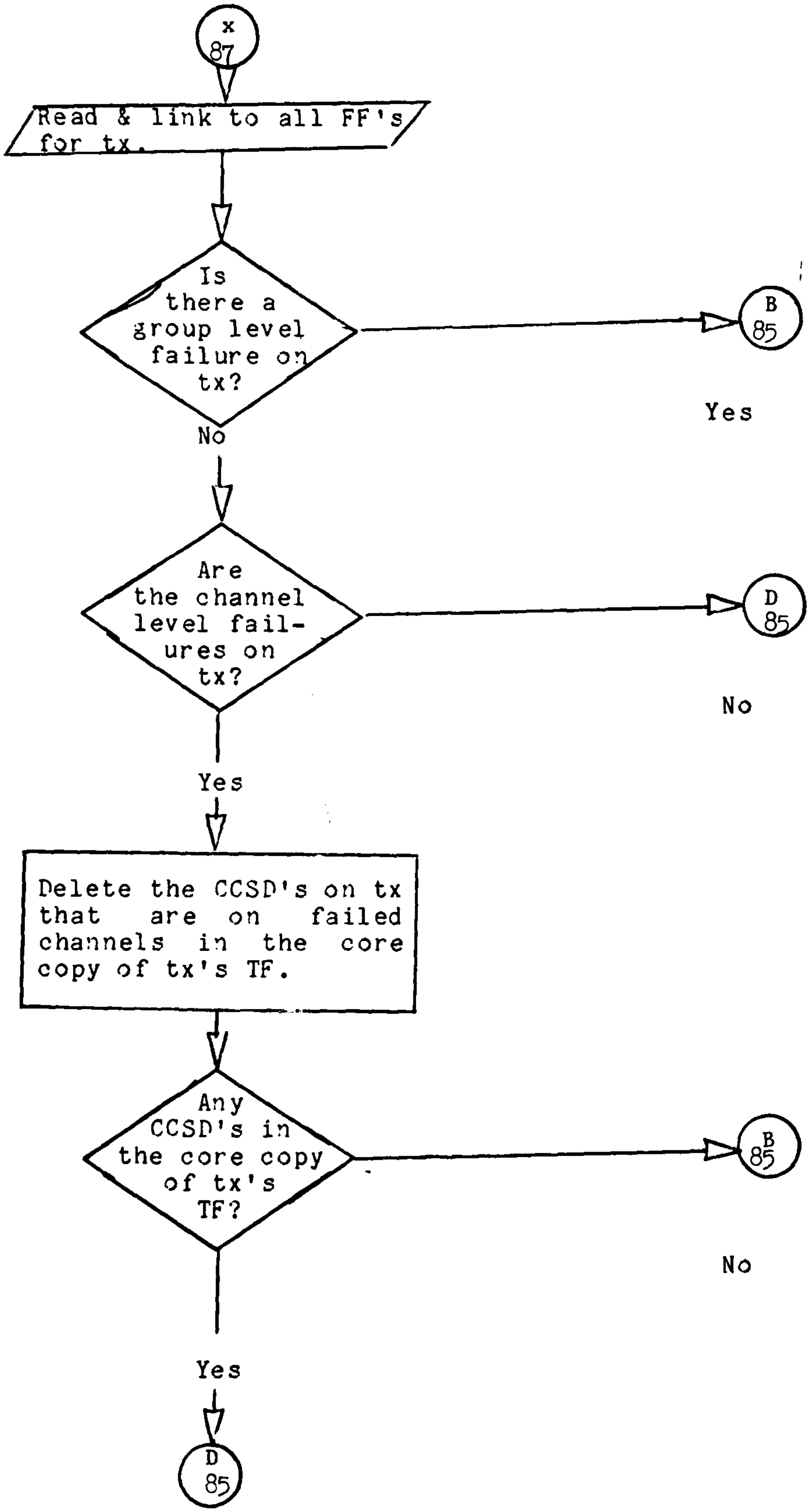

m
82
Is
the pre-emptable
circuit list for
link lx
empty?
Yes
If
lx is a sat-
ellite link, have
any pre-emptable
circuits been
found over
it?
no
r
84
Yes
s
83
No
Select a group from the pre-emptable circuit list for lx with same station end from x.
Open a TREF entry for the other end station of this group. x is the parent station. Note the trunk that was used for the pre-emptable circuits used. Fntries will only be made at this station for circuits that have a pre-emptable circuit to this station. Flag the station as having jumped a goal station if there is a goal station with circuit level access between this station and x.
Delete these circuits from the pre-emptable circuit list for lx.

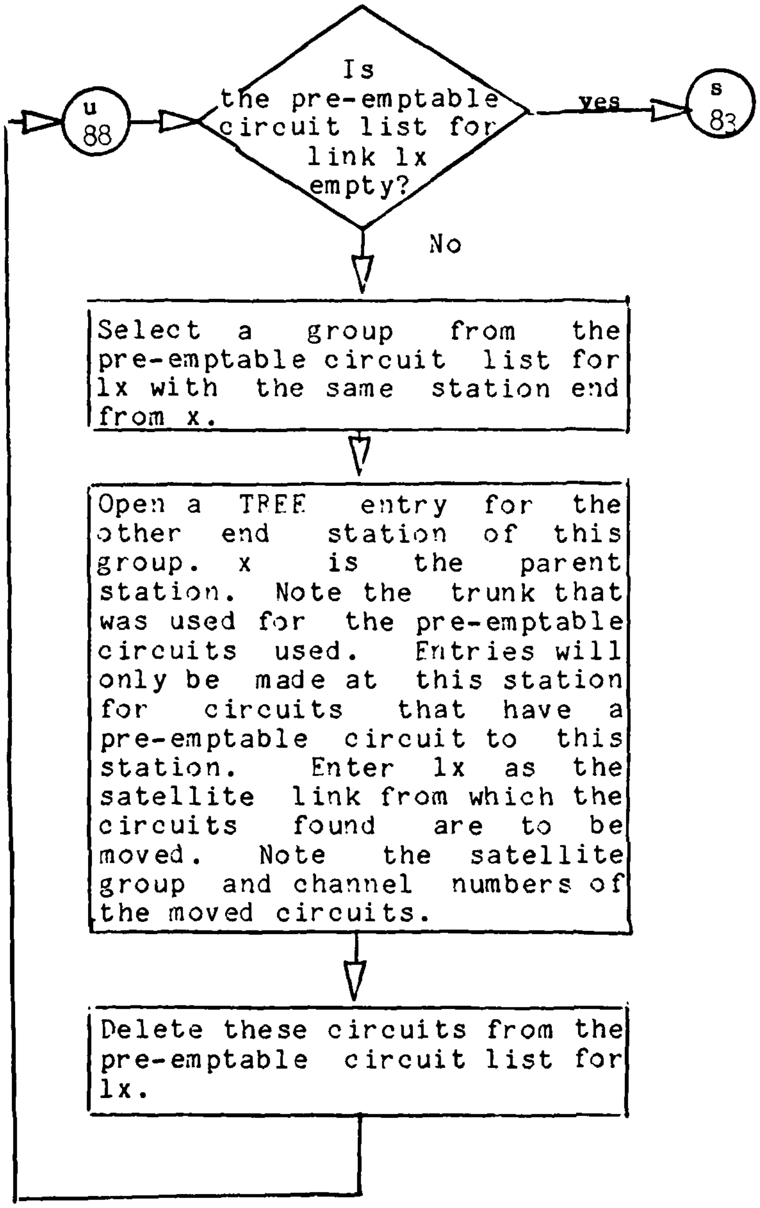

s 83
Any status IP stations in TREE?
No
Delete lx from the link catalog at station x.
Yes
Select an IP entry and calculate COST for the segment from x to this station for the highest RP circuit. Add the cost to x to get total known altroute segment cost.
Is lx a satellite link ?
yes
E 84
no
t 78
For the same entry, calculate HITS & HTS and enter into TREE.
Does any IP entry have the same station as another CLOSED, IP or OPEN entry?
yes
Delete the member of the pair with highest (2*COST+HITS+HTS) for the highest RP circuit entry at that station.
no
Label search status as OPEN for all circuits in the entry having IP status.
Label the remaining member as OPEN for all circuits in the entry.


Read the SF for the station sx on the other end of the satellite link.
r 84
Create a link catalog of satellite links from sx. Do not list totally failed links or the link from x.
Dump the SF of sx.
E 84
Is the link catalog at sx empty?
s 83
yes
No
Read the LF for a link (lx) in the link catalog at x.
Create a catalog of trunks on lx and order them randomly.
Dump the LF for lx.
C 84
Is the trunk catalog empty?
u 88
Yes
No
v 85

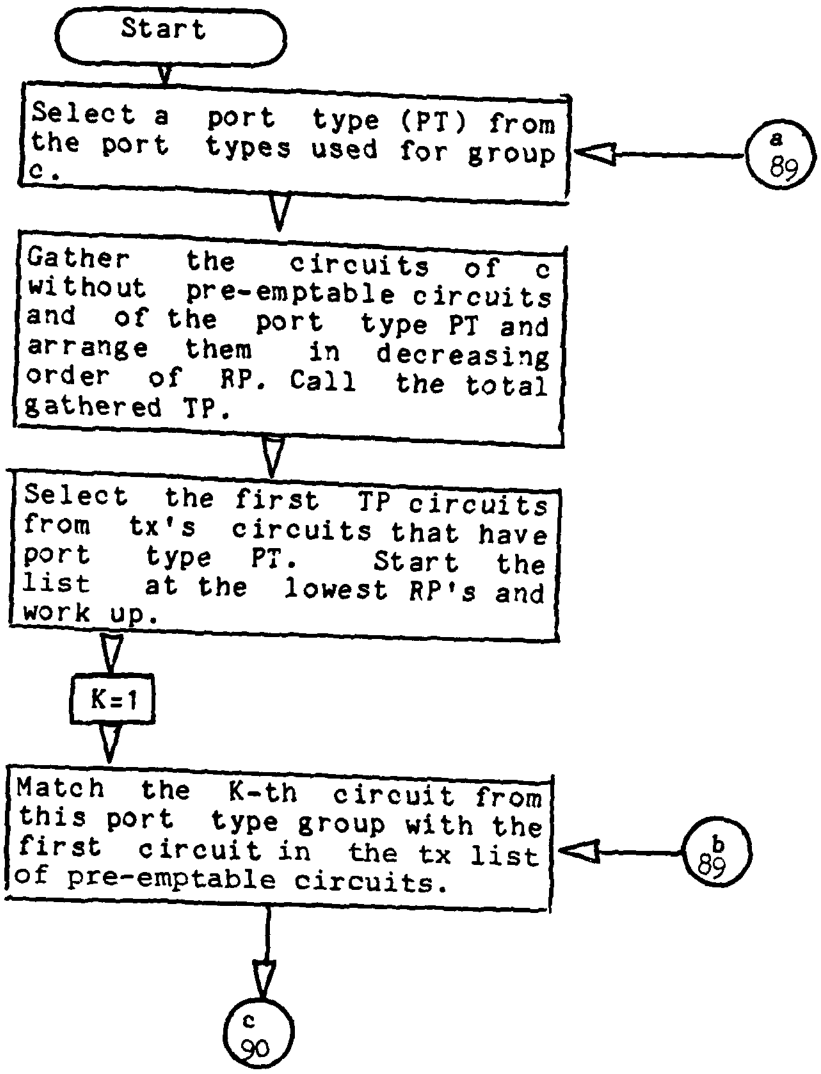

v 85
Read the TF for the top entry in the trunk catalog for lx. (tx)
E 84
A 85
Dump the TF for tx.
Delete tx from the catalog of lx's trunks.
Has tx's other end already been reached via lx trunks for the top M circuits in c?
yes
B 85
no
Any fault files on tx?
x 87
Yes
No
Catalog the station terminating tx's other end over surface links from sx (sy) as having been reached via an lx trunk.
D 85
y 86

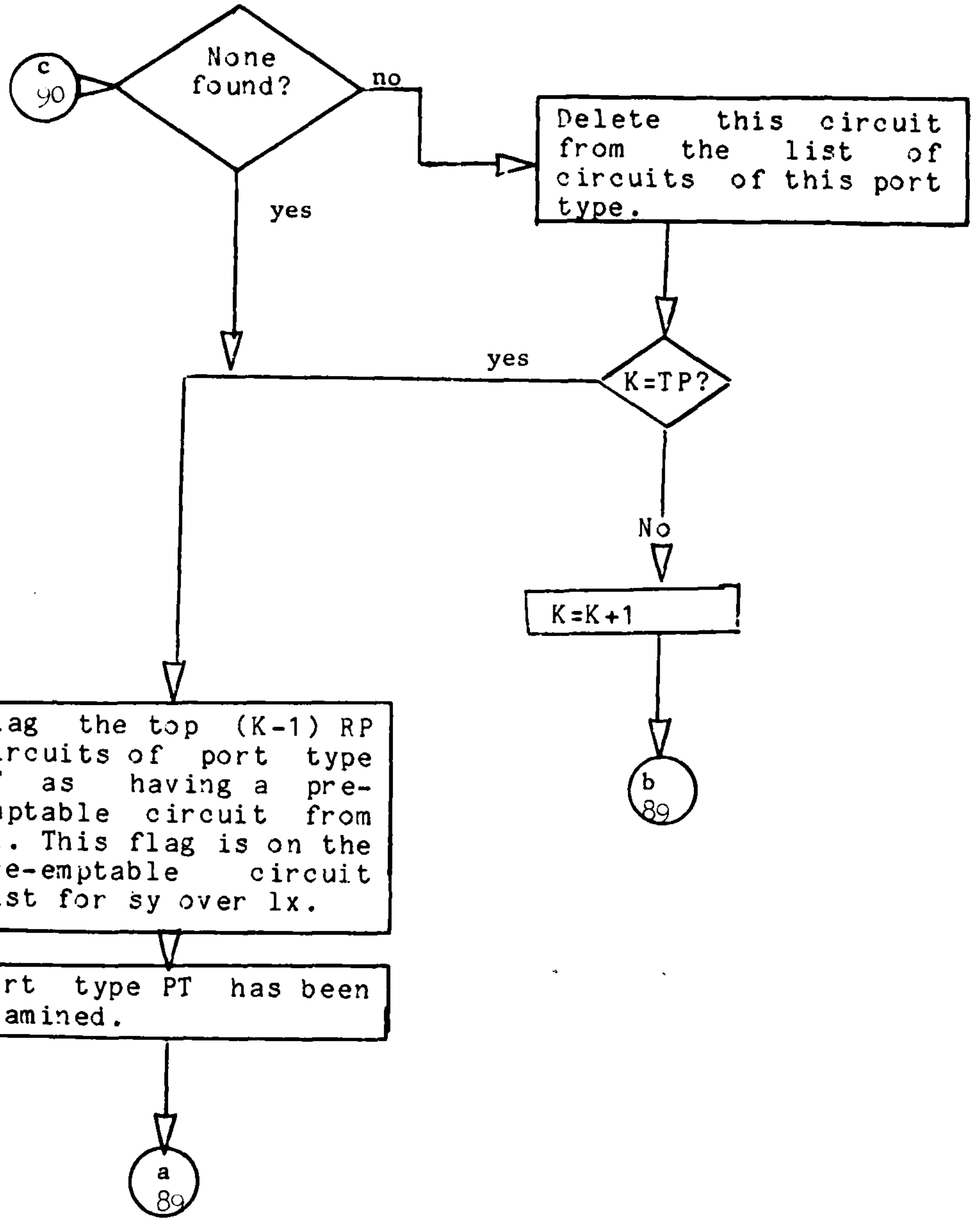

Locate the circuits on tx that have the lowest RP's which are less than c's and the same port types as was used on the normal route. (Use the circuit matching routine for this task.)
y
86
Add to or start a pre-emptable circuit list for sy over link lx. Enter the circuits found to be pre-emptable in the circuit matching routine and note that tx carries these circuits.
Delete tx from the catalog of lx's trunks.
C
84


x
87
Read & link to all FF's for tx.
Is there a group level failure on tx?
Yes
B
85
No
Are the channel level failures on tx?
No
D
85
Yes
Delete the CCSD's on tx that are on failed channels in the core copy of tx's TF.
Any CCSD's in the core copy of tx's TF?
No
B
85
Yes
D
85


u 88
Is the pre-emptable circuit list for link lx empty?
yes
s 83
No
Select a group from the pre-emptable circuit list for lx with the same station end from x.
Open a TREE entry for the other end station of this group. x is the parent station. Note the trunk that was used for the pre-emptable circuits used. Entries will only be made at this station for circuits that have a pre-emptable circuit to this station. Enter lx as the satellite link from which the circuits found are to be moved. Note the satellite group and channel numbers of the moved circuits.
Delete these circuits from the pre-emptable circuit list for lx.

Circuit Matching Routine




c
90
None found?
no
Delete this circuit from the list of circuits of this port type.
yes
yes
K=TP?
No
K=K+1
b
89
Flag the top (K-1) RP circuits of port type PT as having a pre-emptable circuit from tx. This flag is on the pre-emptable circuit list for sy over lx.
Port type PT has been examined.
a
89

2.6.4 The Cost Calculation Routines

2.6.4.1 Discussion of the Routines--As mentioned earlier, the cost factor used in the search routines is the main factor making these routines different from a random search routine. The costs assigned to the stations in the search file TREE must reflect the cost of the partial altroute to the station that has already been found and the estimated cost to a goal station over a route yet unknown. The costs should also be representative of the operator's own weighting of desirable altroute paths. They should be easily calculated and be readily available for modification as operations policy changes. All of these items will be discussed here. The detailed flow charts of the routines deals with the specific implementation.

The costs assigned to each station in the search are the key factors used to determine which station should be examined next for further altroute path expansion. This use of cost directs the search along the most desirable route rather than generating just any or all routes. The desirability of the route is its known cost to the station plus the estimated cost from that station to the goal station. The operations personnel can influence the desirability by controlling the cost of the segments in the network. Thus, the cost assignment routines should not only make use of costs that the tech controllers themselves would use, but allow these costs to be easily reviewed and modified by the operations personnel.

The items chosen to make up a path cost are listed below. They are probably subconsciously used by the tech controller in the current network and could be easily displayed for review and modification.

(1) Route mileage.

(2) The RP's of the circuits pre-empted to use the altroute.

(3) The number of patches made to connect the altroute.

(4) The type of transmission facility used and its relative reliability.

These costs will be used for the segment of the altroute that are known to the station in question. Only some of these costs can readily be estimated for the segment of the altroute not yet found. If the estimates are always less than the actual cost of the segment they cover, then the search routine will always find the lowest cost altroute.

The costs of the known segment of the altroute are found by finding the cost from the parent station of the current station and then adding to the cost found to the parent station.

The altroute mileage is probably going to be the most important cost used. This cost will be the vehicle by which the search routine "sees" the network map and selects a direction to proceed. The calculation of this cost will be simple for the known segment of the altroute - simply add the link mileages of all links used from the starting station of the altroute (FS) to the station being costed. The use of the Connectivity Path File (CNF) will make this cost readily available to operations personnel and make it available in one place for easy modification. Every path in the CNF will have the links and link mileages for paths in the DCS area. The mileages between the current station and its parent station are simply added up along the paths found in the CNF. For cases where the paths do not touch, the euclidean distance between the ends of the paths is added to the path mileages from the CNF. For cases where one station does not lie on a path, the euclidean distance from that station to the closest path end of the path the other station lies on is added to the path mileage. If neither station lies on a path, simply find the euclidean mileage between stations. This requires a new data file called the Station Coordinate File (SCF) to store the coordinates of all area stations. More is said about the structure of this new file in the next section.

The estimation of the mileage of the unknown altroute segment can be found in a similar way by examining the current station in the search and the goal station of the altroute (TS or ITS).

The pre-empted circuit's RP's is the second cost considered. Obviously, it is desirable to pre-empt as few and as lowly circuits as possible for the altroute. This cost tallies both considerations. To find this cost for the segment that is known, simply add the RP's of pre-empted circuits along the search. Some numerical numbers should be assigned to the currently used RP's so as to allow numerical manipulation along with the other costs.

The estimation of this cost for the unknown segment of the altroute does not seem practical and will not be used.

The number of patches required for an altroute gives an indication of the effort required to implement the altroute and the number of trunks used in tandem. This again is readily known for the segment already found, but is not estimated for the unknown segment.

The final cost is the transmission facility cost. The CNF proposed here should carry a transmission cost for every link which reflects the cost of that facility relative to other types of transmission available and relative to other facilities of the

same type. This allows unreliable links to be avoided in the search unless there is no alternative route. Again, this is an excellent operations input to search control.

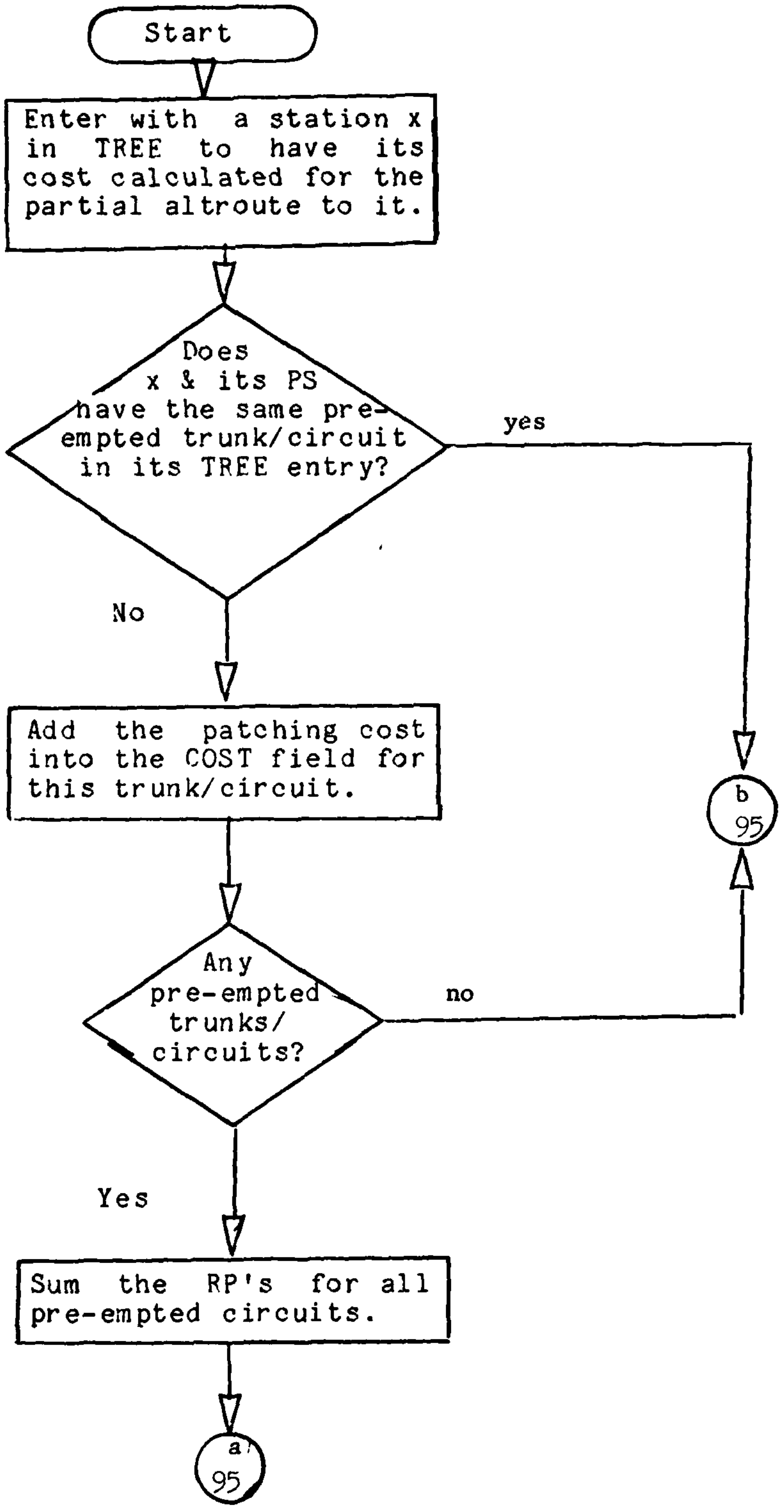
The calculation of this cost proceeds similar to the mileage costs which are also extracted from the CNF. The transmission costs over distances not on a path can be estimated by mutiplying the euclidean distance by some average transmission cost per mile.

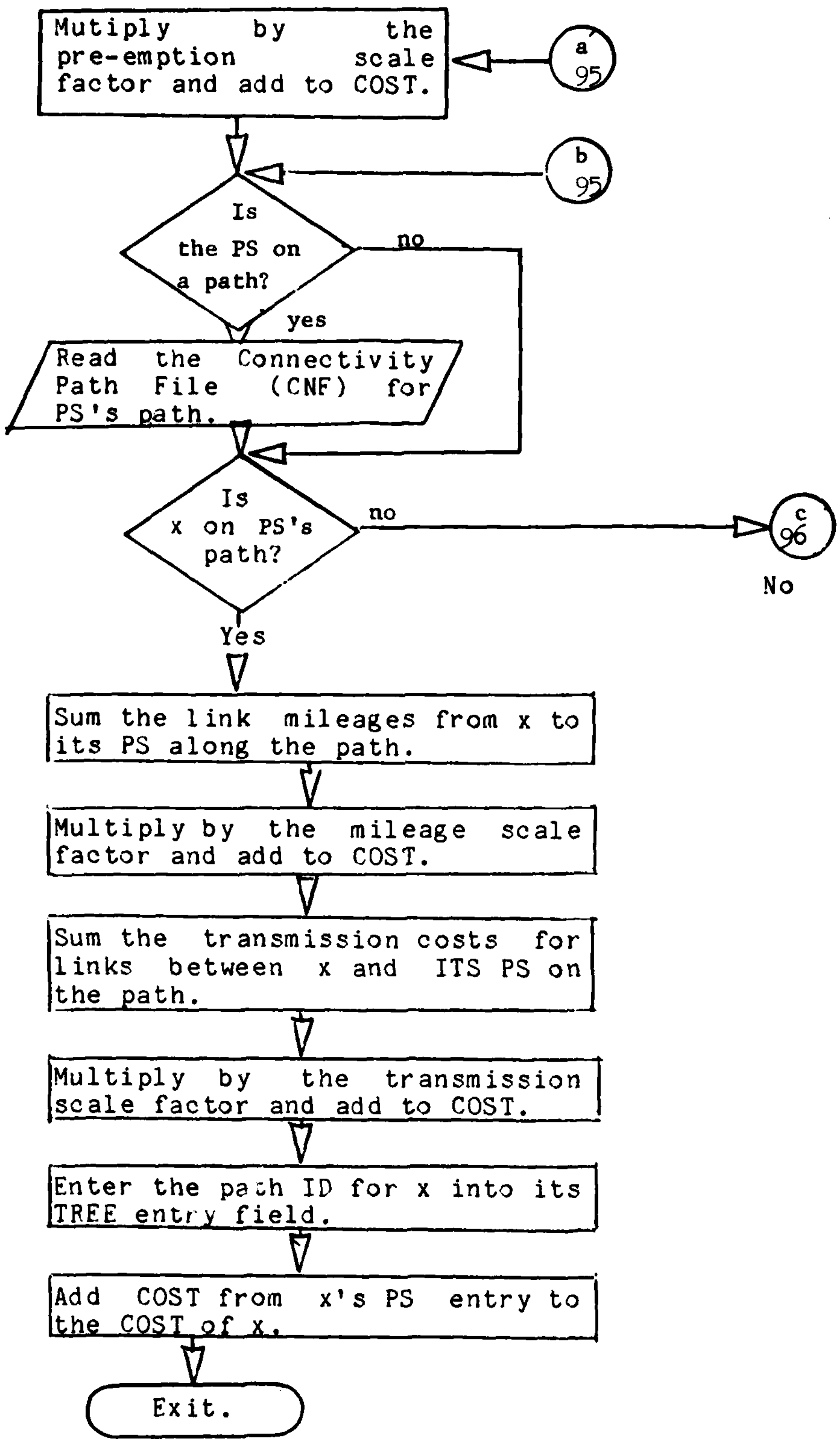
The total cost of the segment to the station or from the current station to the altroute goal is found as a weighted sum of these calculated and estimated costs. The scale factors used to weight the individual terms will determine the importance given each cost. This is an input from operations personnel to control the search routine and can be varied to reflect policy changes. The best way to establish these factors is certain to be actual running of the algorithms and studying the results as compared to tech controller or operations altroute selection.

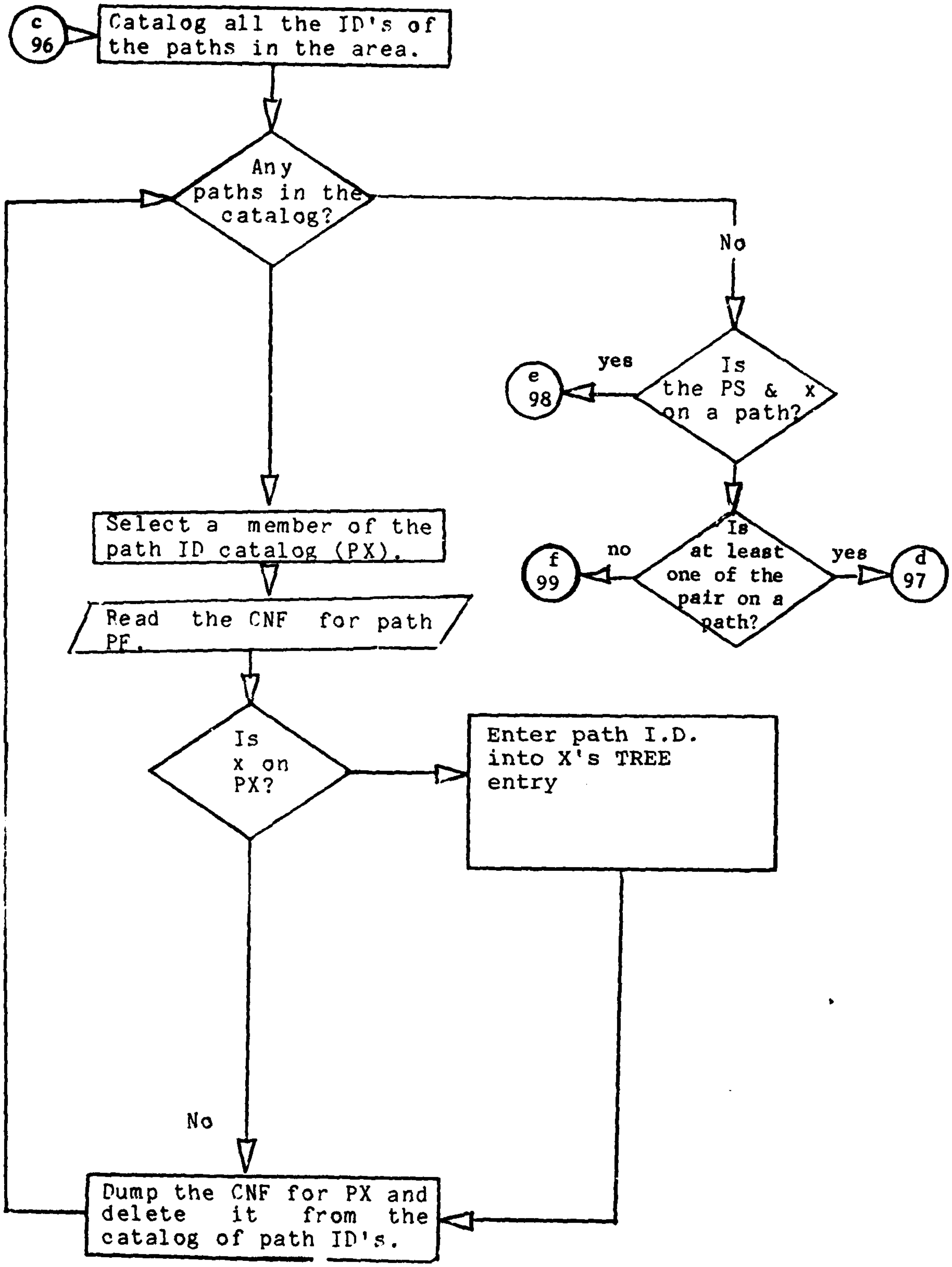
2.6.4.2 Required Data Base--The routine requires the station whose costs are to be calculated to have a fully completed TREF entry. The CNF and SCF are then the other required files to be used. The CNF was part of the original data base but has had link mileages and transmission costs added as shown in appendix A.

The Station Coordinate File (SCF) is a new file being proposed here for cost calculation. The station coordinates could be read from the SF if a new field were added but this would spread out this data and require reading many files rather than just one when distance calculations are involved. This file might also be interesting to controllers as part of their CRT data base. This file would list every station I.D. with the coordinates of the station relative to some local area reference. This would require 9 bytes or a total of 900 bytes for the European network. This is a small file if one considers the big job it does, especially in mileage estimation over the unknown altroute segment.

2.6.4.3 Flow Chart of the Cost Calculating Routine--




Mutiply by the pre-emption scale factor and add to COST.
a 95
b 95
Is the PS on a path?
no
yes
Read the Connectivity Path File (CNF) for PS's path.
Is x on PS's path?
no
c 96
No
Yes
Sum the link mileages from x to its PS along the path.
Multiply by the mileage scale factor and add to COST.
Sum the transmission costs for links between x and ITS PS on the path.
Multiply by the transmission scale factor and add to COST.
Enter the path ID for x into its TREE entry field.
Add COST from x's PS entry to the COST of x.
Exit.


c
96
Catalog all the ID's of the paths in the area.
Any paths in the catalog?
No
Is the PS & x on a path?
yes
e
98
Is at least one of the pair on a path?
no
f
99
yes
d
97
Select a member of the path ID catalog (PX).
Read the CNF for path PF.
Is x on PX?
Enter path I.D. into X's TREE entry
No
Dump the CNF for PX and delete it from the catalog of path ID's.

d
97

Use the Station Coordinate File (SCF) to find the euclidean distance from the station without a path to the nearest end of the path the other station (on a path) is on. Note that nearest path end station.

Sum the link mileages from that path end to the station on that path. Add to the euclidean distance found above.

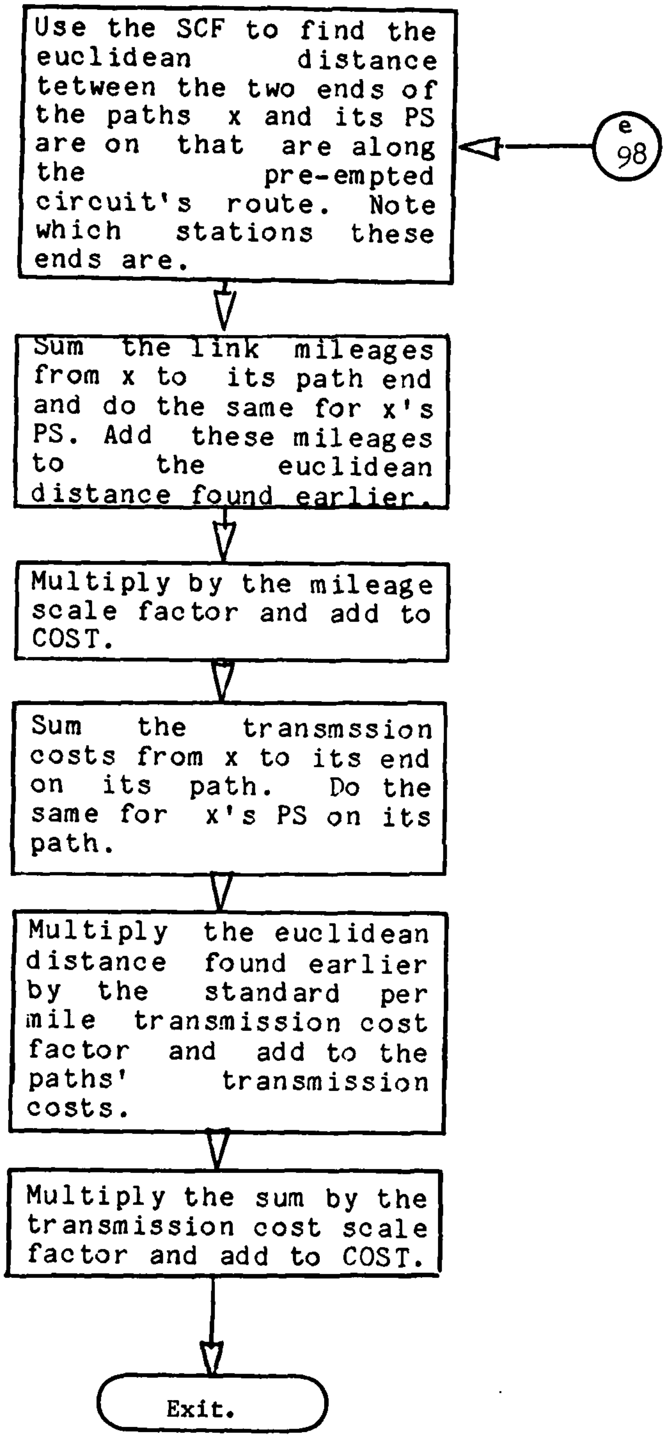
Multiply by the mileage scale factor and add to COST.

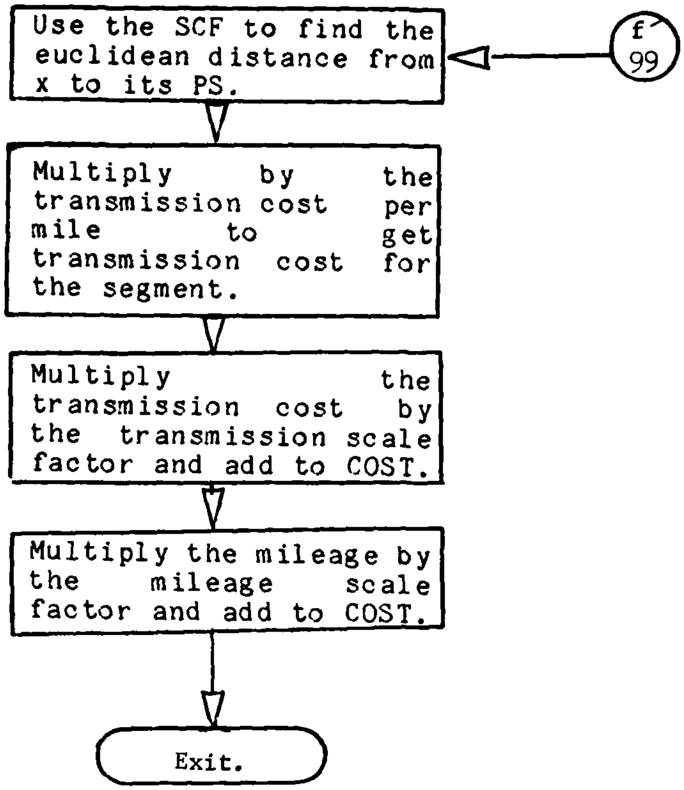
Sum the transmission costs for the links from the path end to the station on that path.

Multiply the euclidean distance found above by the standard transmission cost per mile and add to the transmission cost on the path.

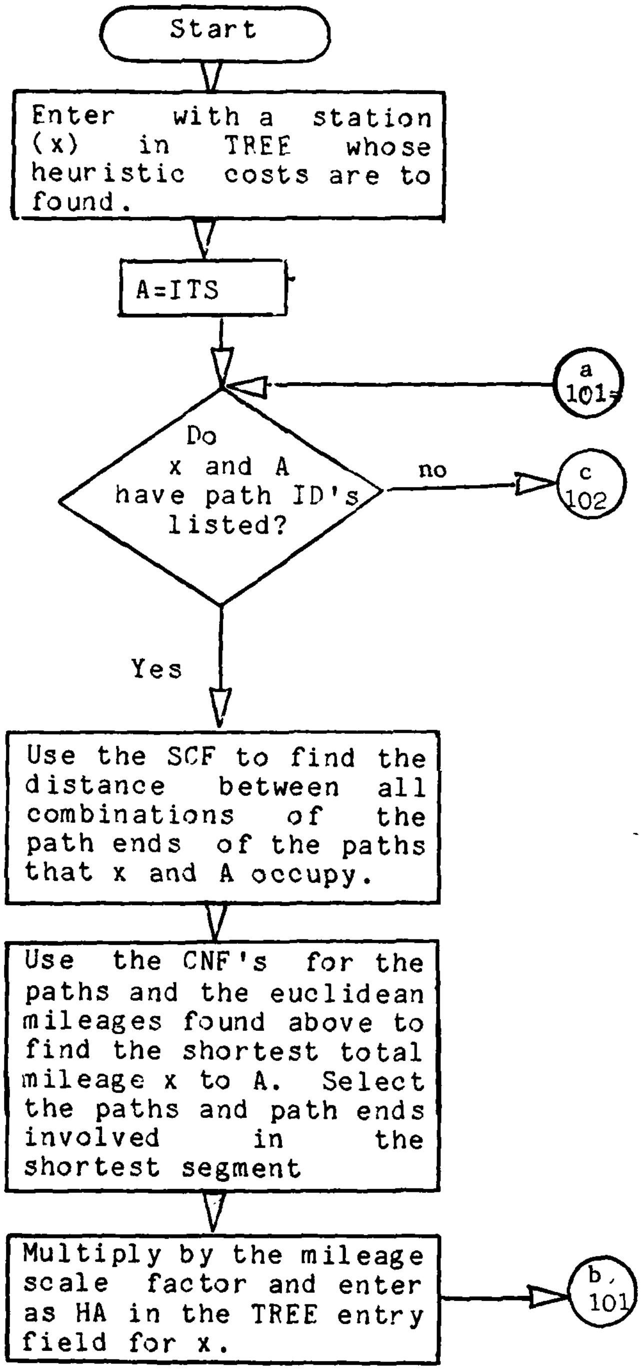
Multiply by the transmssion cost scale factor and add to COST.

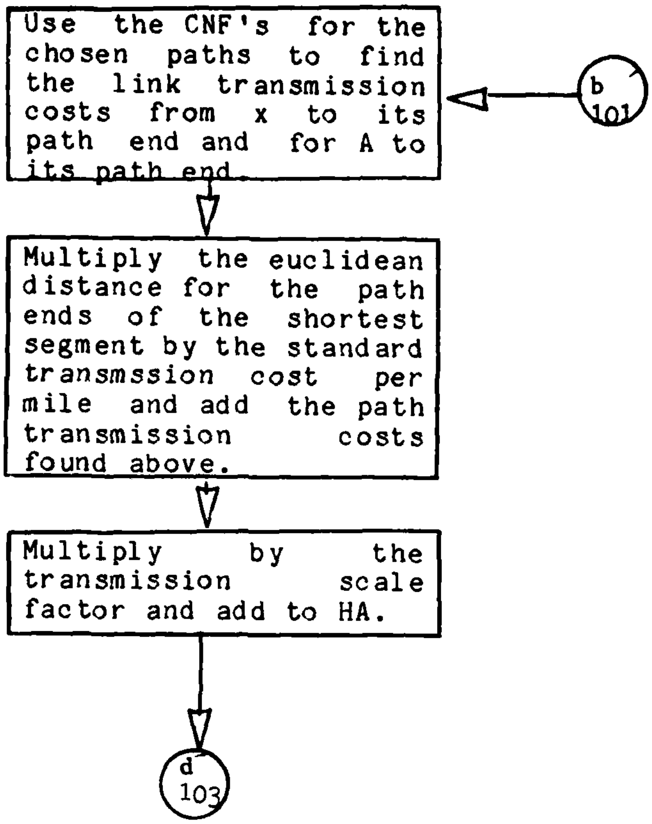
Exit.

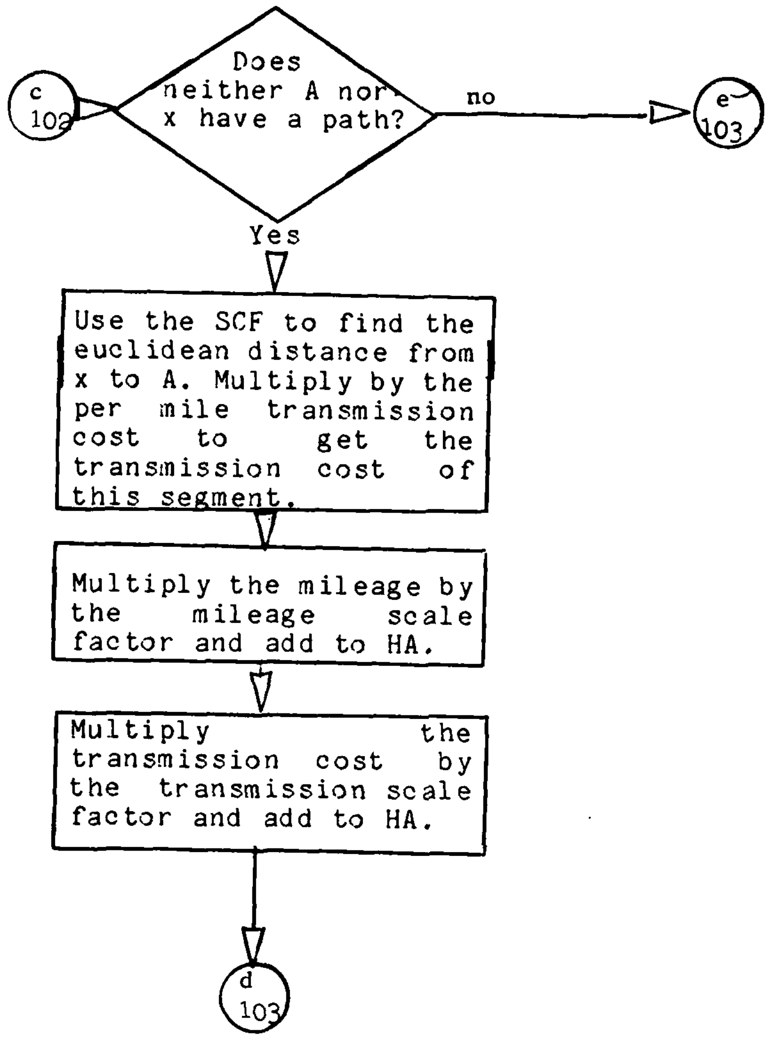

Use the SCF to find the euclidean distance tetween the two ends of the paths x and its PS are on that are along the pre-empted circuit's route. Note which stations these ends are.
e
98
Sum the link mileages from x to its path end and do the same for x's PS. Add these mileages to the euclidean distance found earlier.
Multiply by the mileage scale factor and add to COST.
Sum the transmssion costs from x to its end on its path. Do the same for x's PS on its path.
Multiply the euclidean distance found earlier by the standard per mile transmission cost factor and add to the paths' transmission costs.
Multiply the sum by the transmission cost scale factor and add to COST.
Exit.

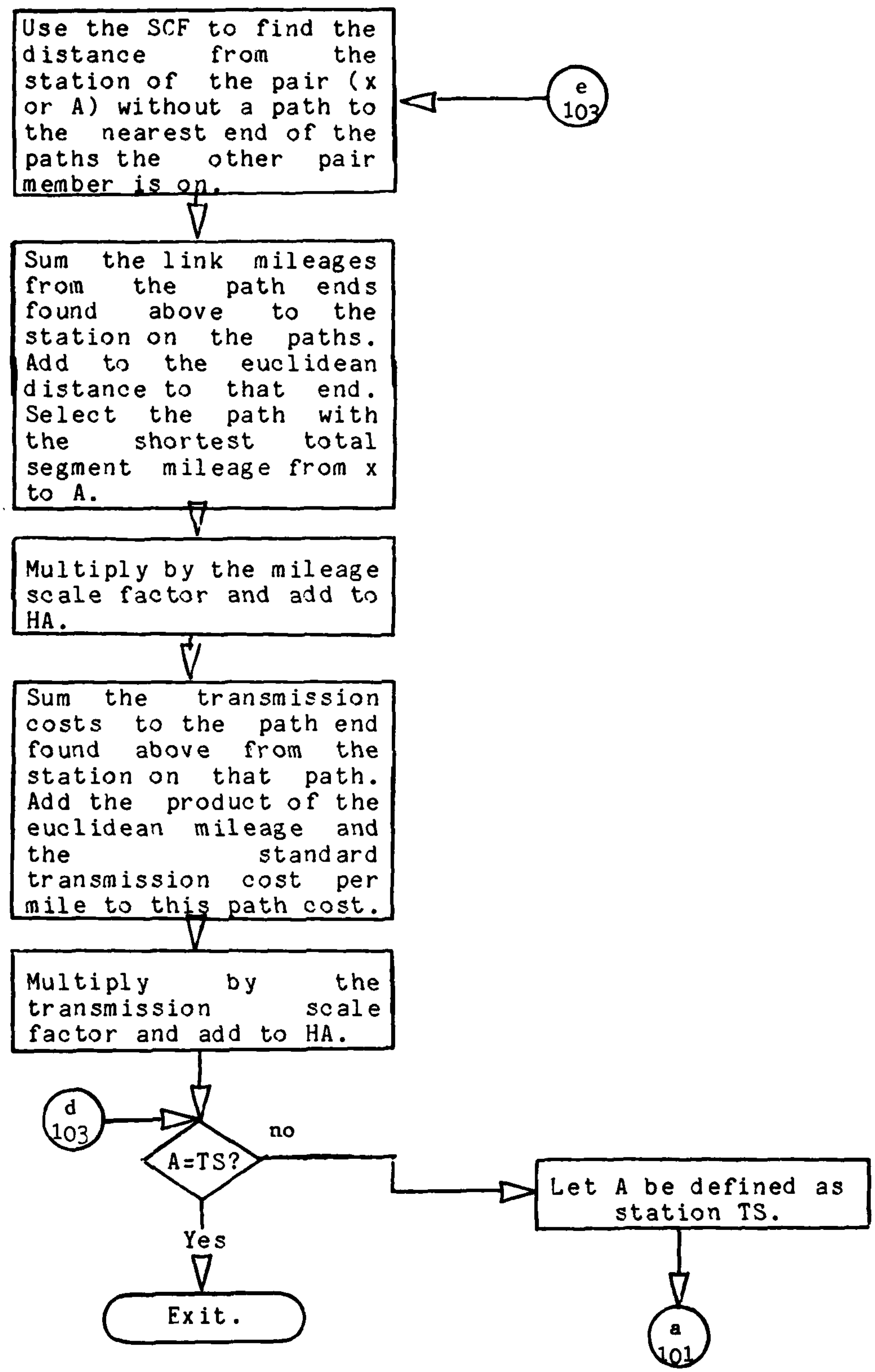

f
99
Use the SCF to find the euclidean distance from x to its PS.
Multiply by the transmission cost per mile to get transmission cost for the segment.
Multiply the transmission cost by the transmission scale factor and add to COST.
Multiply the mileage by the mileage scale factor and add to COST.
Exit.

2.6.4.4 The Heuristic Cost Calculation Routine--




Use the CNF's for the chosen paths to find the link transmission costs from x to its path end and for A to its path end.
b
101
Multiply the euclidean distance for the path ends of the shortest segment by the standard transmssion cost per mile and add the path transmission costs found above.
Multiply by the transmission scale factor and add to HA.
d
103


c 102
Does neither A nor x have a path?
no
e 103
Yes
Use the SCF to find the euclidean distance from x to A. Multiply by the per mile transmission cost to get the transmission cost of this segment.
Multiply the mileage by the mileage scale factor and add to HA.
Multiply the transmission cost by the transmission scale factor and add to HA.
d 103


Use the SCF to find the distance from the station of the pair (x or A) without a path to the nearest end of the paths the other pair member is on.
e 103
Sum the link mileages from the path ends found above to the station on the paths. Add to the euclidean distance to that end. Select the path with the shortest total segment mileage from x to A.
Multiply by the mileage scale factor and add to HA.
Sum the transmission costs to the path end found above from the station on that path. Add the product of the euclidean mileage and the standard transmission cost per mile to this path cost.
Multiply by the transmission scale factor and add to HA.
d 103
A=TS?
no
Let A be defined as station TS.
a 101
Yes
Exit.

2.6.5 The Goal Station Definition Routine

2.6.5.1 Discussion of the Routine--The goal station definition routine does the job of defining the goal stations for the search routines and also finds a suitable starting point for the search along with the identification of the ends of the service. The terminating end of the longest segment of the intact portion of the circuit/trunk with a failure is called the "originating station" (OS). The other end of this segment from which the search begins is the "first station" (FS). The terminating end of the shortest of the remaining intact segment of the service is the "terminal station" (TS). Its other end which bounds the group of stations which the search routine attempts to reach is called the "intermediate terminal station" (ITS). The goal stations are any stations with the proper multiplex level access to the service being altrouted that lie between TS and ITS. Thus, two intact segments of the original service's route are defined to bound the segments of the service that has failures.

The search routine that finds the lowest cost altroute should bridge the smallest part of the failed trunk/circuit. Thus, the routine of goal definition starts by isolating the first stations of the proper mux level before the failed stations segment as FS and ITS. Normally these will bound the two intact segments to be linked by an altroute. Figure 2-5 shows this simple case of segment identification. Stations A,B,C and D define the OS/FS segment with station D as FS. The triangles at the stations denote proper mux. level access for the service being altrouted. Station D is the first station with access to the service to bound the failure of link D-F from the OS terminal. The other segment is the ITS/TS segment of stations G and F. Since station E did not have access to the proper mux. level, station F was chosen to be ITS. The search routine now starts at station D and searches until either station F or G is reached.

Figure 2-6 points out a case where selection of the station nearest a totally failed link is not a good choice for FS or ITS. Here the altroute would have to "backhaul" to B before finding an alternate route around the failure. Also note that getting to F is not desirable because that forces a "backhaul" to get to G. The routine finds these cases by looking for single operational links from the last station on the FS/OS segment. Finding such a situation forces the FS station back to a point (station B) where another link is found that leaves the dead end that the failed link created. Notice that the selection of ITS follows the same guideline. Note again that the D-E link is totally failed, not just failed in the channel or group carrying the altroute circuit. If link D-E had other groups, then FS and ITS would have been identified as figure 2-5.

The case shown in Figure 2-7 indicates that if the last station bounding the failure does have at least two operational links, then the original FS choice is desirable to keep the altroute

short. Note that D rather than C determined that the B-C-D segment was not a dead end. In figure 2-8,C examination shows that B-C-D is a dead end and that B should be selected as FS to begin the search. This occurs even though D has no access to the service being altrouted.

If, in the process of moving FS back towards OS, the OS station is actually reached, some special questions are asked by the routine. If OS has only one operational link from it, we know that there is no station on the OS/FS segment that appears to have access to the service and provide another link from this dead end segment. (See Figure 2-9). If there are no stations from OS to the failed segment that have 3 or more operational links, then OS is isolated on a dead end segment and no altroute is possible. Failure of the routine is noted and no altroute search is begun. On the other hand, as in figure 2-9, the station B has 3 links but no access to the service, OS will be selected as FS to start the search. The reason for this choice is that some pre-emptable circuit from A may have access at B and provide the needed link away from the seemingly dead end segment. This same strategy is used in the selection of ITS for a similar case on the other intact segment.

If a dead end segment is discovered as in figures 2-6, 2-8 and 2-9, then the link B-C is temporarily put on the failed link list. This prevents the search from looking down the fruitless D-C-B path .

The stategies discussed above apply to trunks and circuits when goal station definition is required.

2.6.5.2 Goal Station Definition for Trunks--The routine flow charted here is for circuit level altrouting. The routine for trunks is not given because it is so similar to the circuit level routine. The trunk routine would begin by identifying the stations of circuit or group level access as trunk level access stations. From there on, replacing the words circuit level access with trunk level access will make the routine do goal definition for trunks.

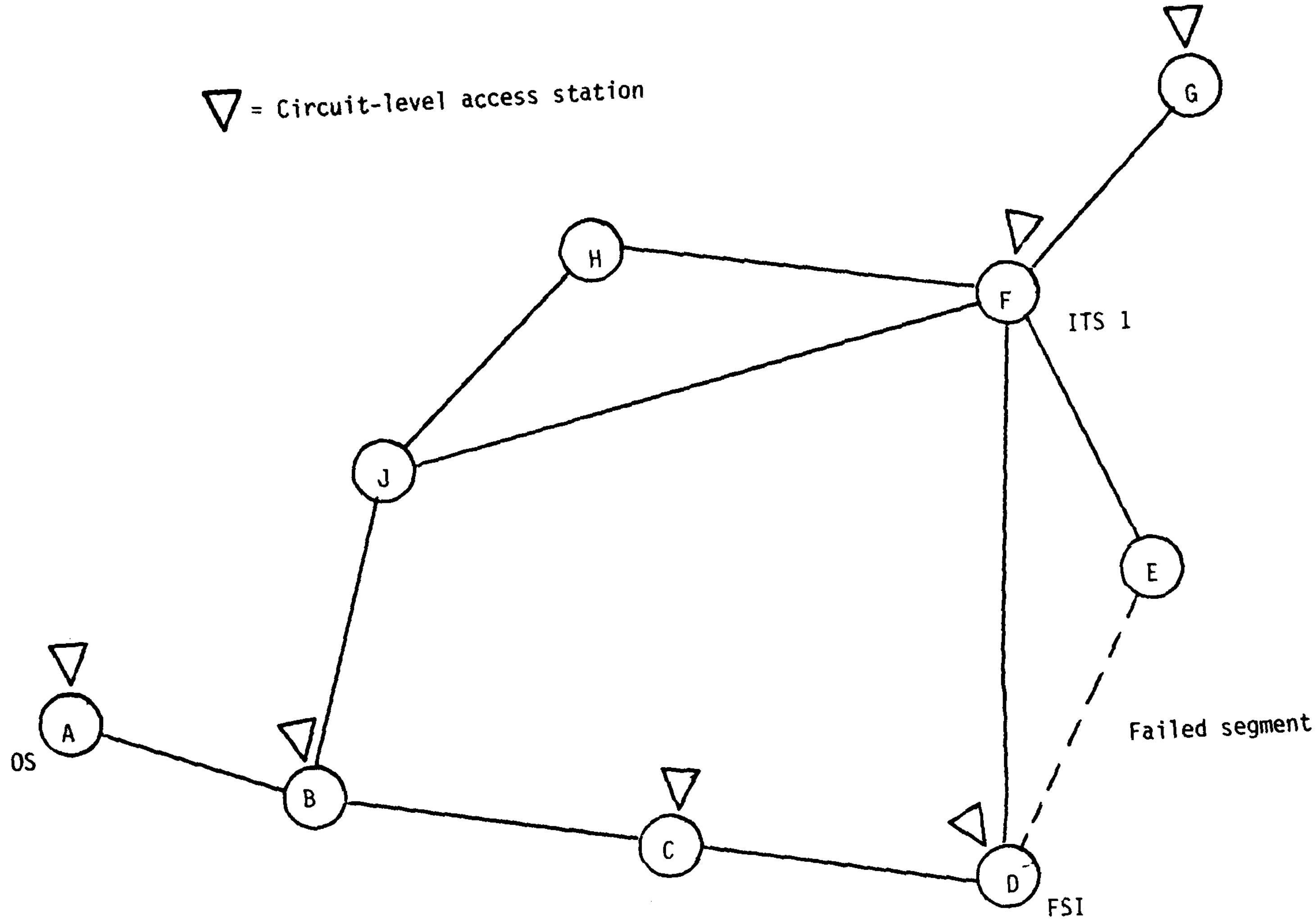


Figure 2-5: Typical intact segment definition

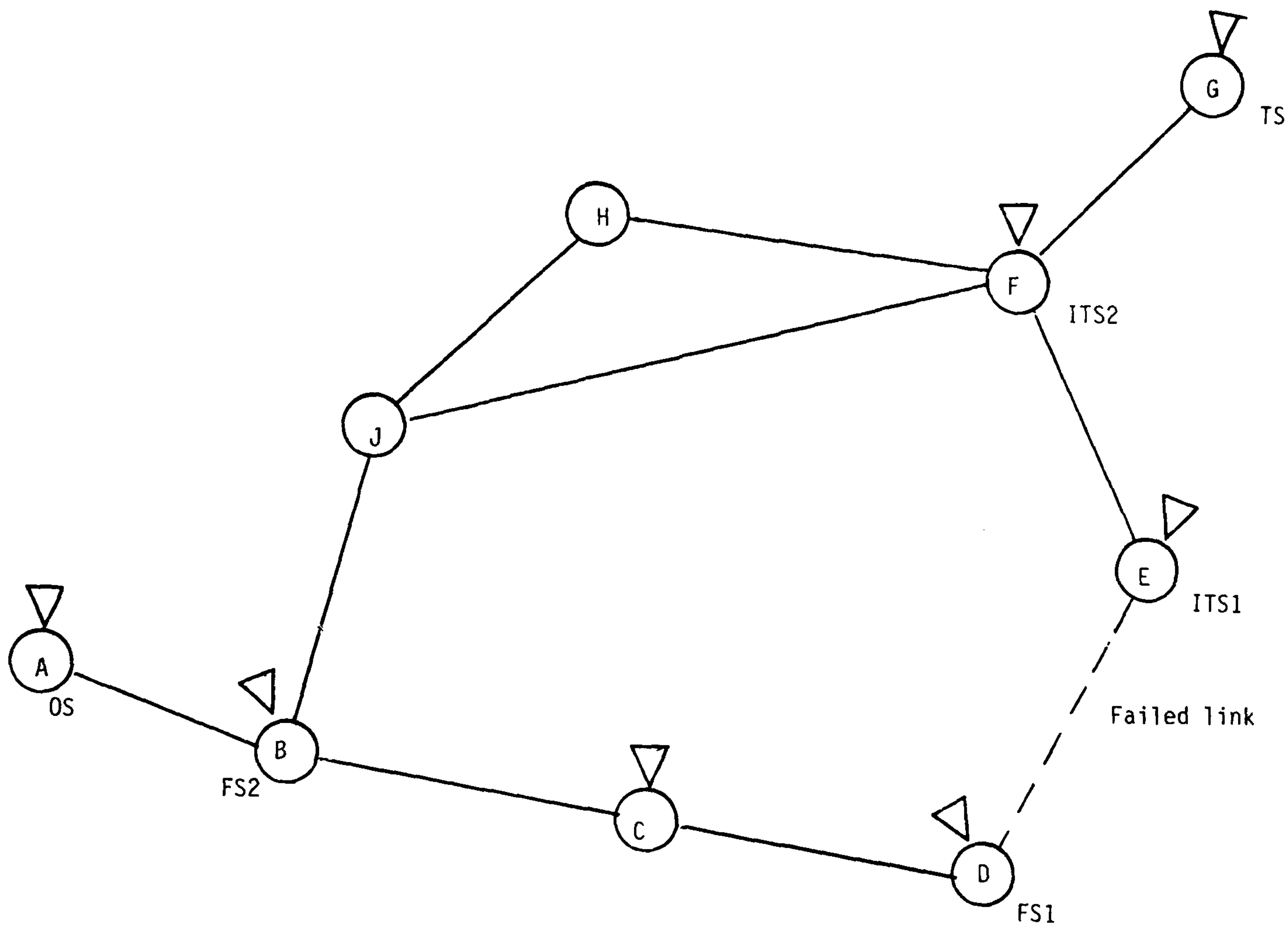


Figure 2-6: Intact segment definition when link has failed to create a dead end spur

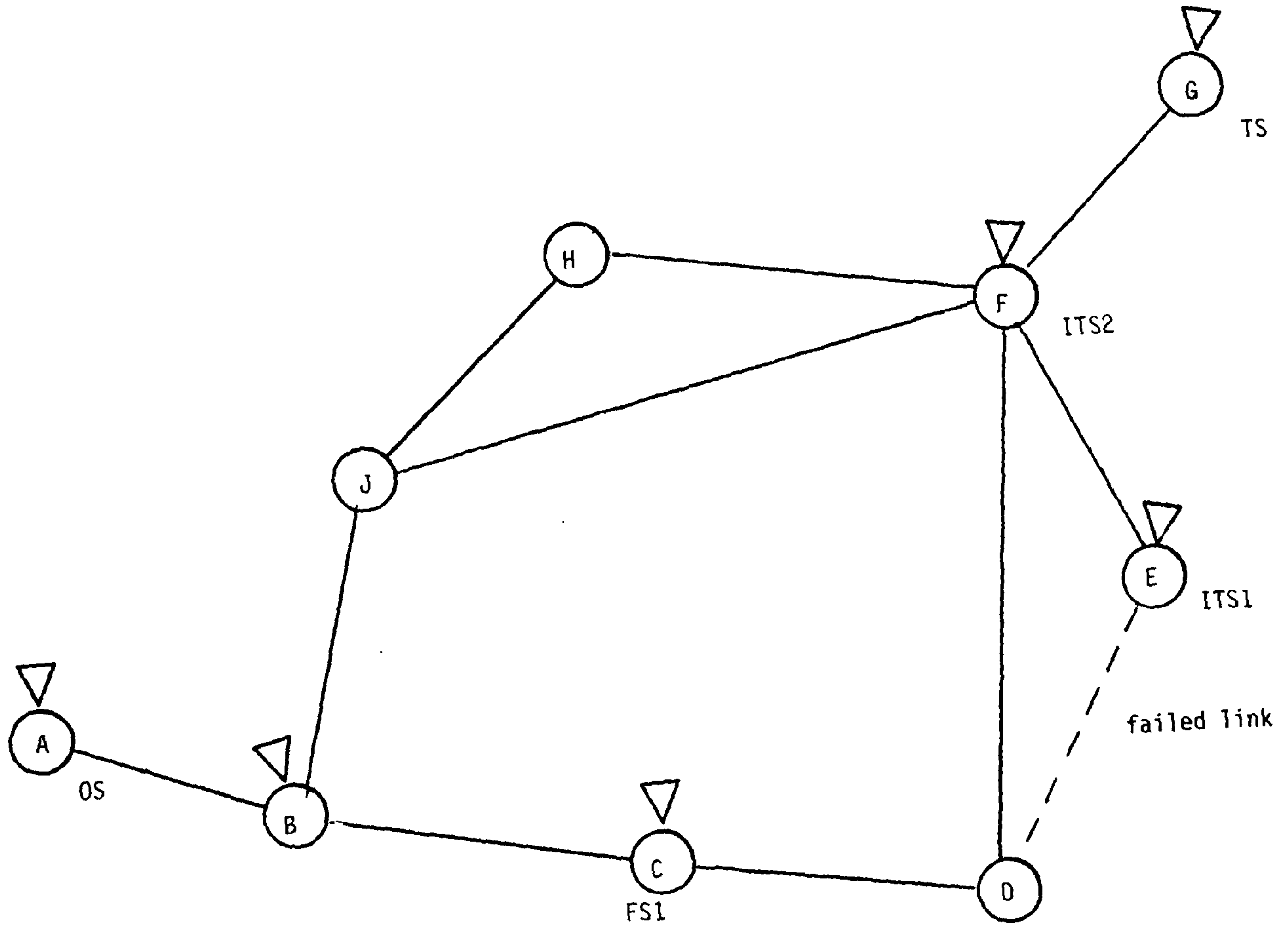


Figure 2-7: Intact segment definition for a failed link without a dead end spur

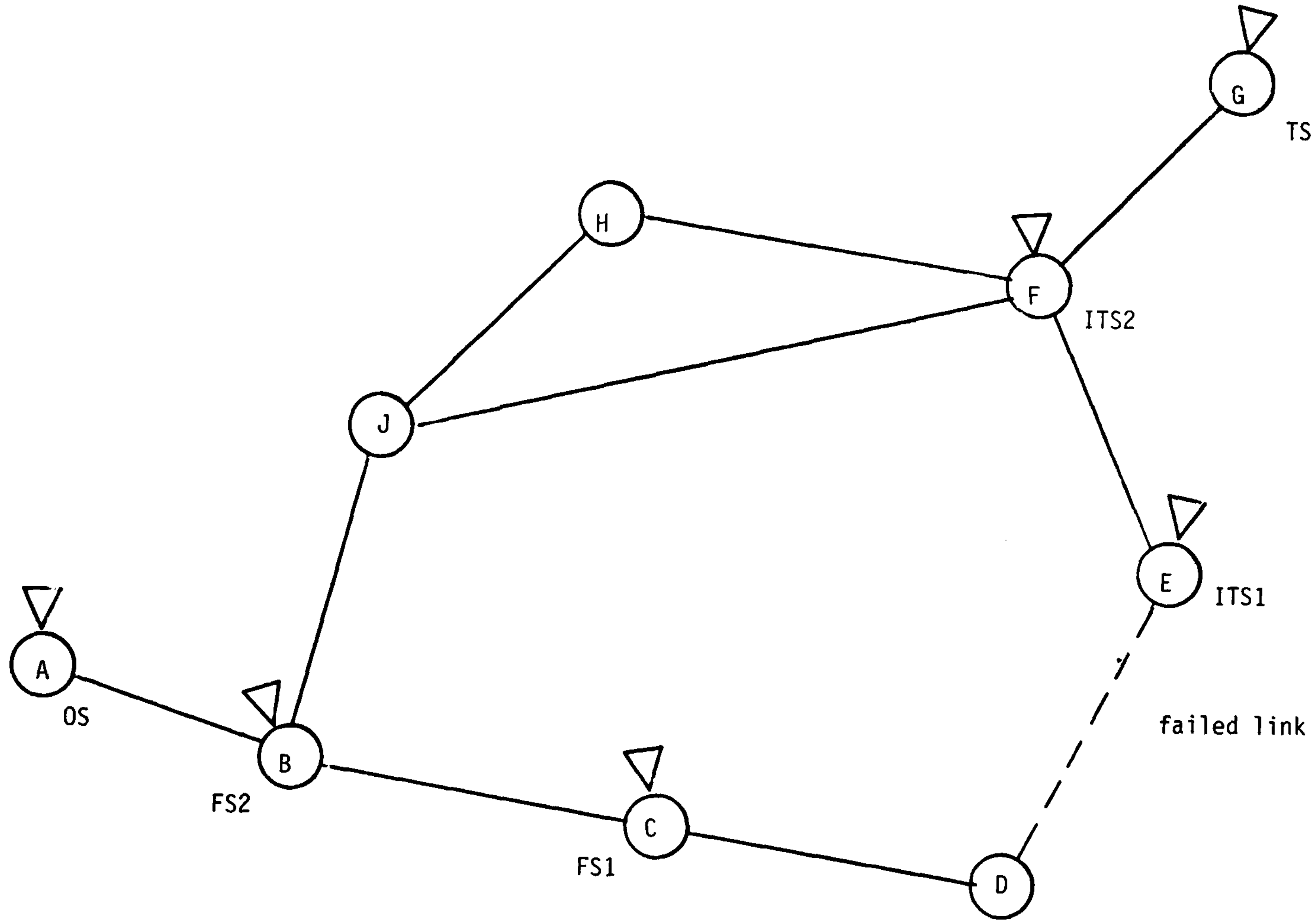


Figure 2-8: Intact segment definition with a failed link creating a dead end spur

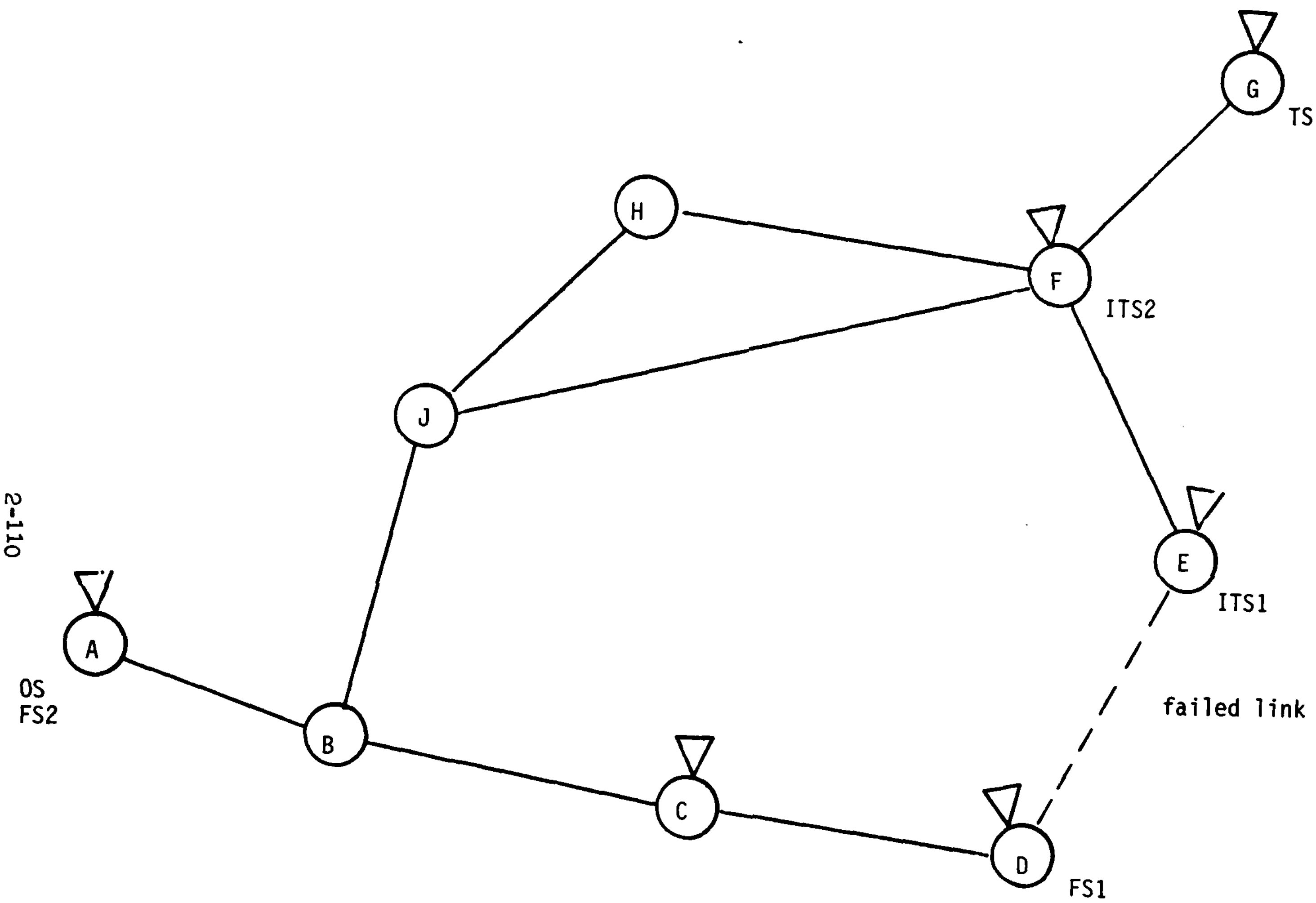
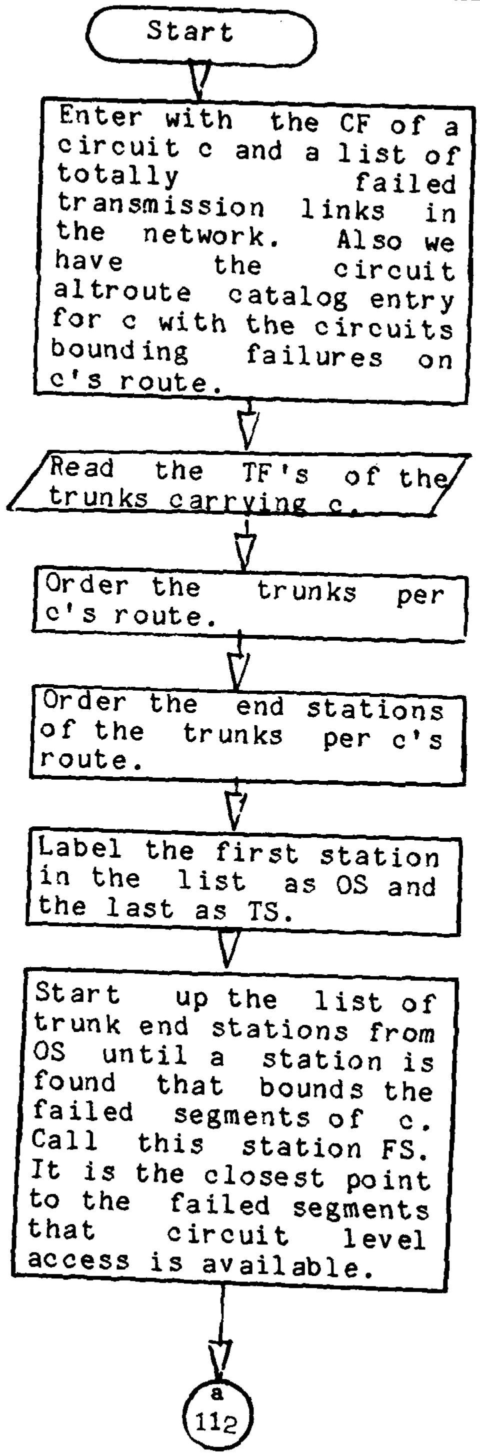
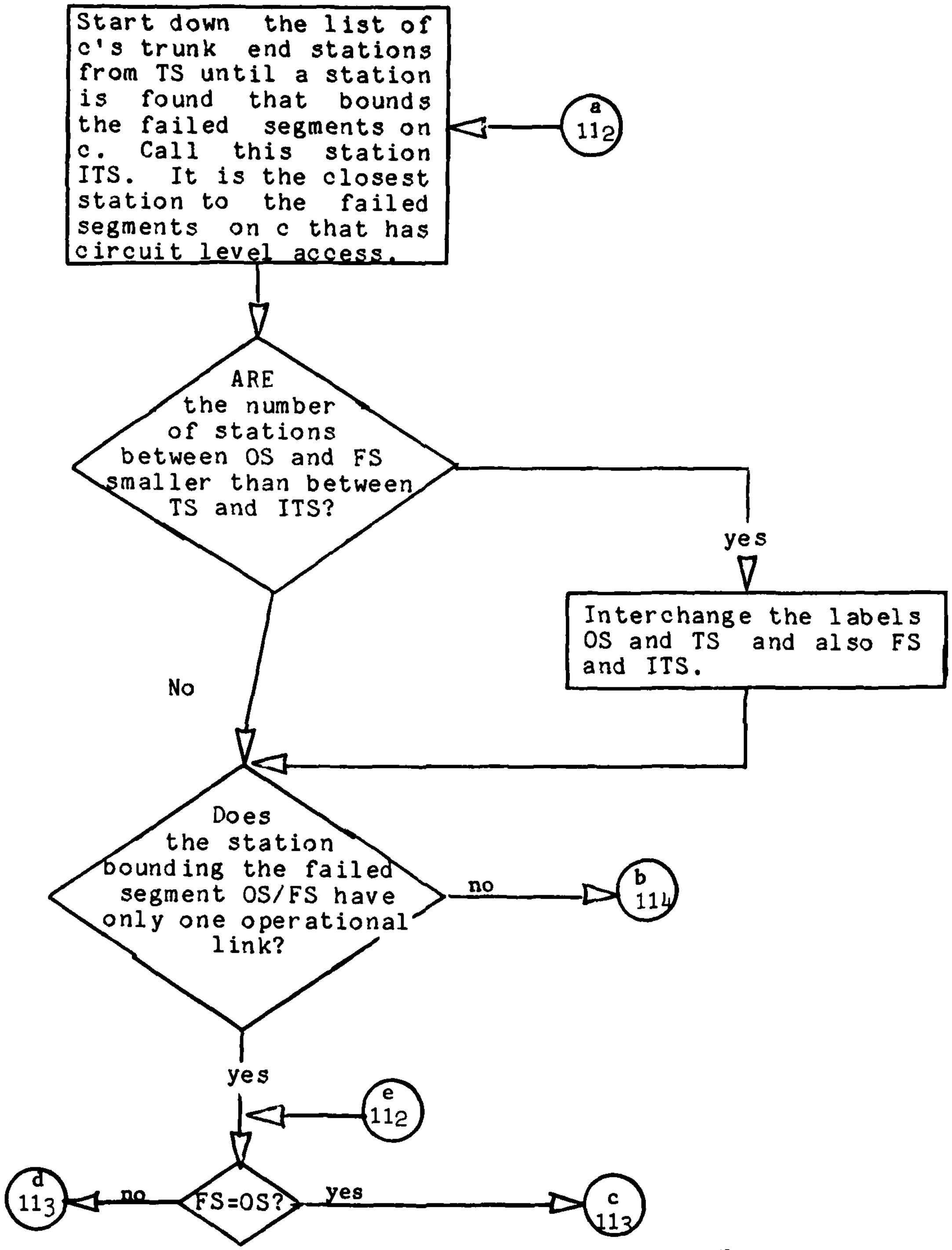


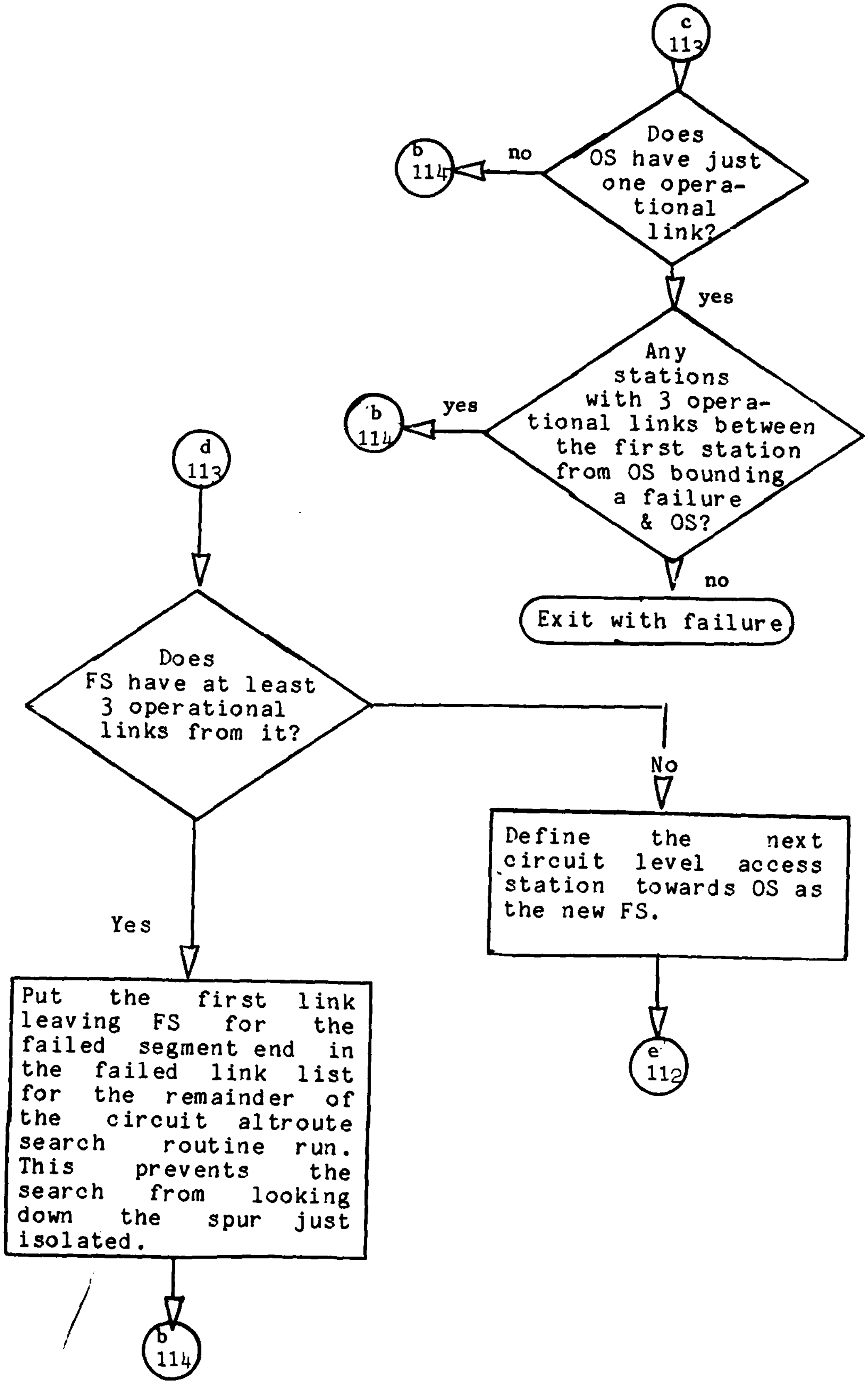
Figure 2-9: Intact segment definition with a failed link and only secondary access off of the dead end spar

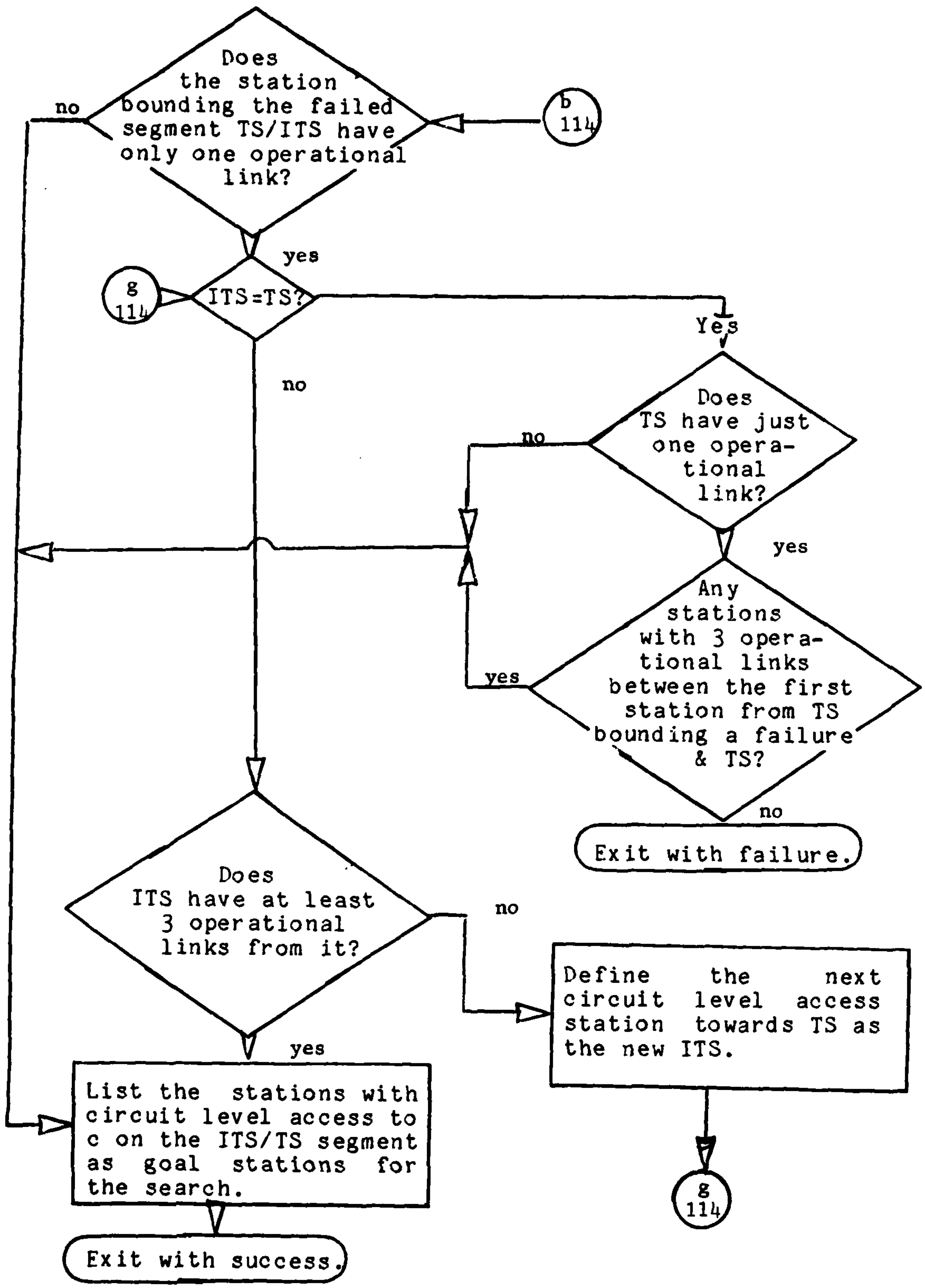
2.6.5.3 <u>The Goal Station Definition Routine</u>--




Start down the list of c's trunk end stations from TS until a station is found that bounds the failed segments on c. Call this station ITS. It is the closest station to the failed segments on c that has circuit level access.
a
112
ARE the number of stations between OS and FS smaller than between TS and ITS?
yes
Interchange the labels OS and TS and also FS and ITS.
No
Does the station bounding the failed segment OS/FS have only one operational link?
no
b
114
yes
e
112
d
113
no
FS=OS?
yes
c
113

Yes


c 113
Does OS have just one operational link?
no
b 114
yes
Any stations with 3 operational links between the first station from OS bounding a failure & OS?
yes
b 114
no
Exit with failure
d 113
Does FS have at least 3 operational links from it?
Yes
Put the first link leaving FS for the failed segment end in the failed link list for the remainder of the circuit altroute search routine run. This prevents the search from looking down the spur just isolated.
b 114
No
Define the next circuit level access station towards OS as the new FS.
e 112



Does the station bounding the failed segment TS/ITS have only one operational link?
no
b 114
yes
g 114
ITS=TS?
Yes
no
Does TS have just one operational link?
no
yes
Any stations with 3 operational links between the first station from TS bounding a failure & TS?
yes
no
Exit with failure.
Does ITS have at least 3 operational links from it?
no
yes
Define the next circuit level access station towards TS as the new ITS.
g 114
List the stations with circuit level access to c on the ITS/TS segment as goal stations for the search.
Exit with success.

2.6.6 The Restoral Routine

2.6.6.1 Discussion of the Routine--The restoration of the normal route of a circuit or trunk is carried out by the restoral routine described in this section. The routine is reached by two different means. For trunks or circuits which have been disrupted by equipment failure, the removal of the last fault file linked to that service will trigger an input to this routine. The second way to enter the routine is the removal of all pre-empted segments of a circuit or trunk. Any service which is pre-empted will have a pre-emption listed in its data file and be placed on both the altroute and restoral catalog. The circuits and trunks on the restoral catalog are reviewed when pre-emptions in any part of the network are removed to check for the pre-emption-free state. Thus, the catalog of trunks and circuits awaiting restoral is key in this routine just as the altroute catalog was in the altroute search routines. The detailed flow chart of the routine is given in section 2.6.6.2, but we will discuss the main features here.

The routine is activated by the deletion of the last fault file to some circuit or trunk. This action means that the service disrupted is now restorable. It also means that any altroute used can be dismantled and allow restoral of the services that it pre-empted. Without a stimulus of equipment repair, we know that the circuits or trunks on the restoral catalogs still have pre-emptions and cannot be restored. Thus, no action is taken by this routine in between reports of repaired equipment.

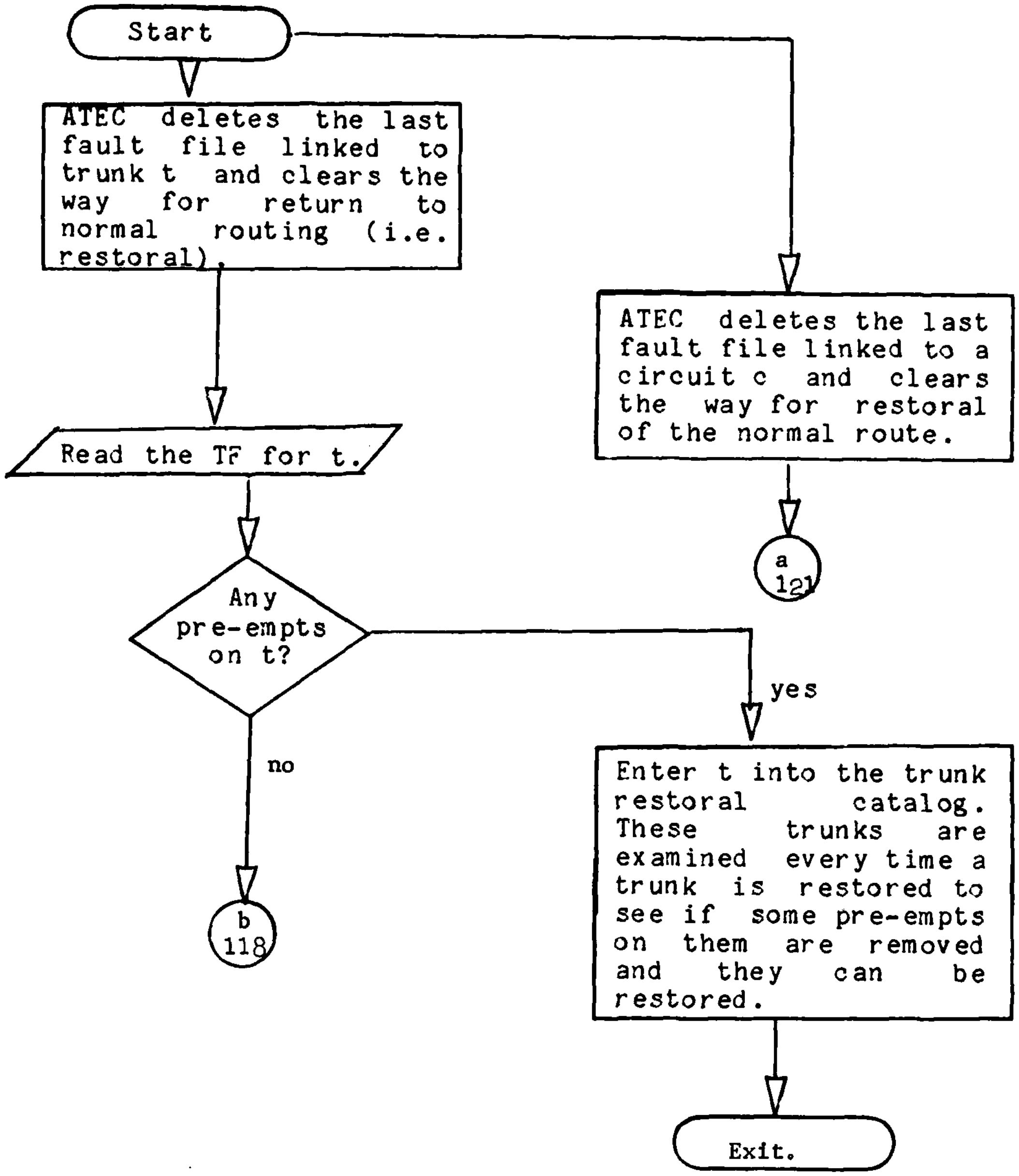
If the trunk (we will select this case to discuss in detail, the circuit case is identical) entered from the trigger of fault file removal has no pre-emptions, it can be restored immediately to normal routing. If there are any pre- emptions on it, it must be set aside (in the trunk restoral catalog) to await the removal of its pre-emptions. The reason for this was explained earlier - any other trunk pre-empting this trunk must have had higher RP's and thus should not be pre-empted to restore this trunk.

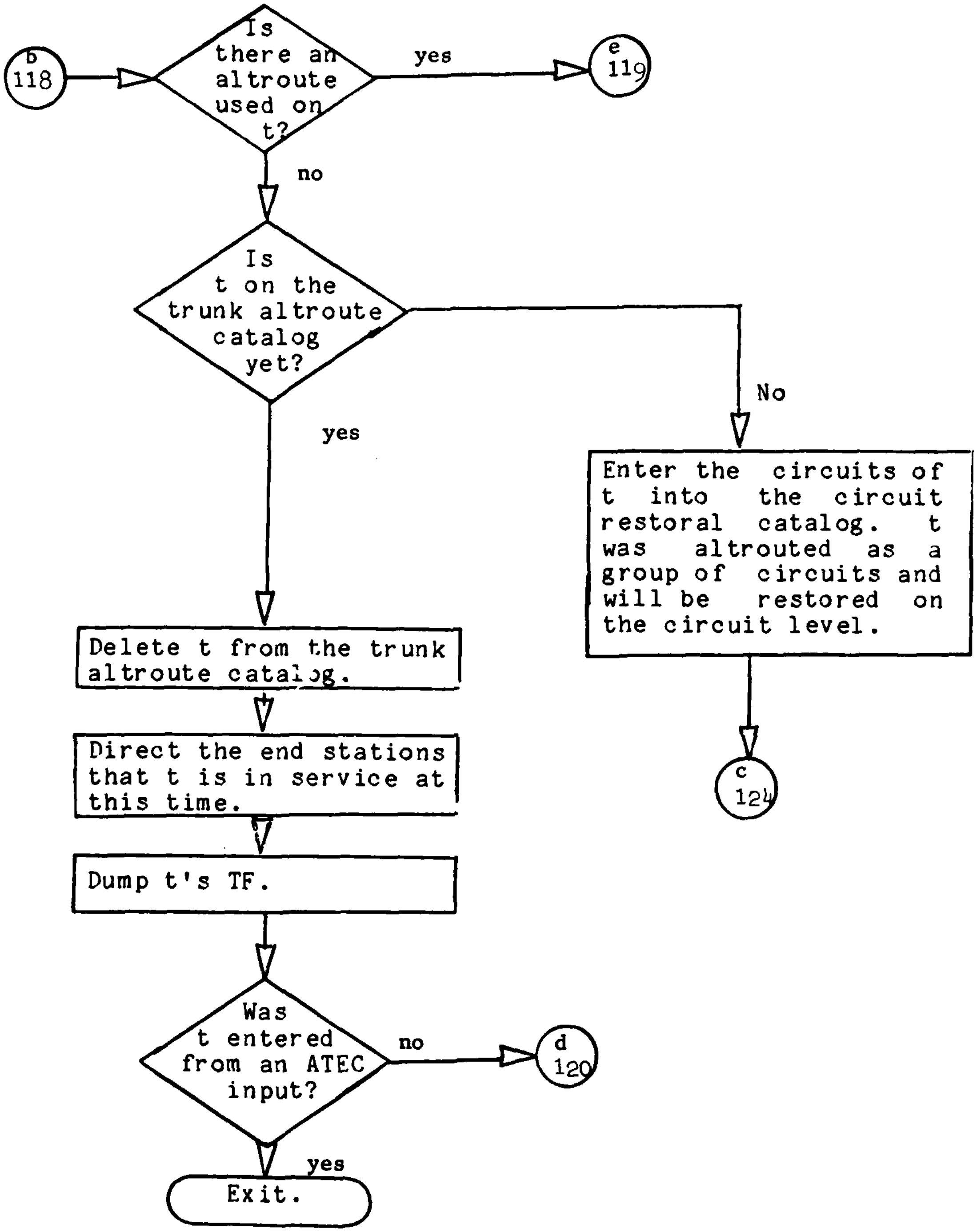
If the trunk can be restored, the next question is whether or not it has been altrouted? If it has not, then restoral is simply deleteing it from the restoral catalog and informing the users that service is restored.

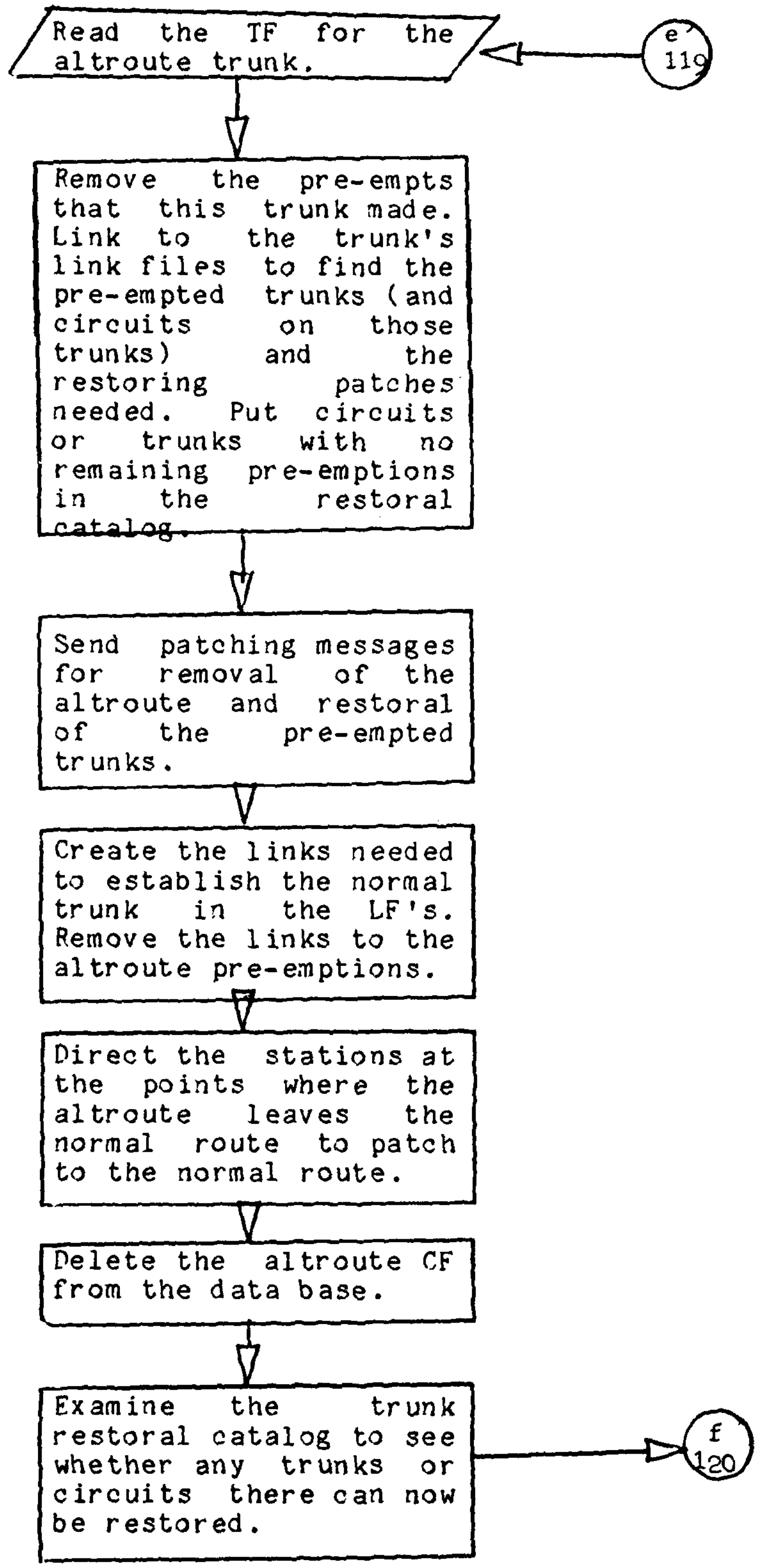
A restorable trunk that has an altroute is restored with a little more activity. The altroute's pre-emptions must be removed. Appendix A gets into the details of the pre-emption data base links that must be removed. Suffice it to say that trunk and circuit pre-emptions are removed. If the last pre-emption on a trunk or circuit is removed in this process, then that service should be put on the restoral catalog so the formal notification of users and altroute removal can take place. The removal of the altroute requires unpatching the route and repatching to the segment of the normal route that was failed.

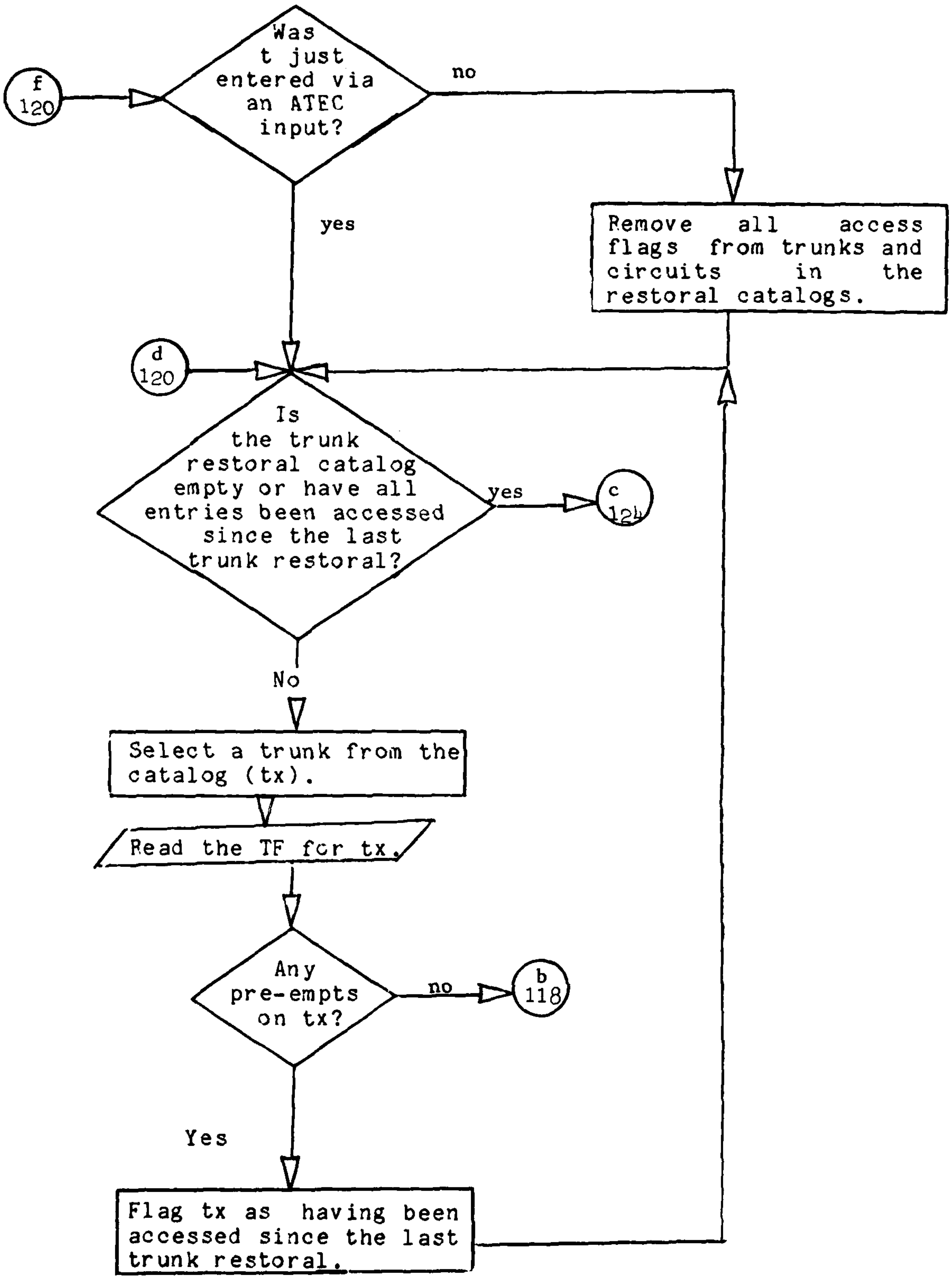
The last thing to do if an altroute was removed and pre-empted services released is to check those circuits or trunks that are in the restoral catalog and restore those that now have no pre-emptions. These services are entered into the restoral routine just like those entered via a repair report. Both trunks and circuits are checked for pre-emption-free status. This is done because decomposed trunks may be stored as circuits awaiting restoral and the removal of a pre-empting trunk frees the circuits now in the restoral catalog.

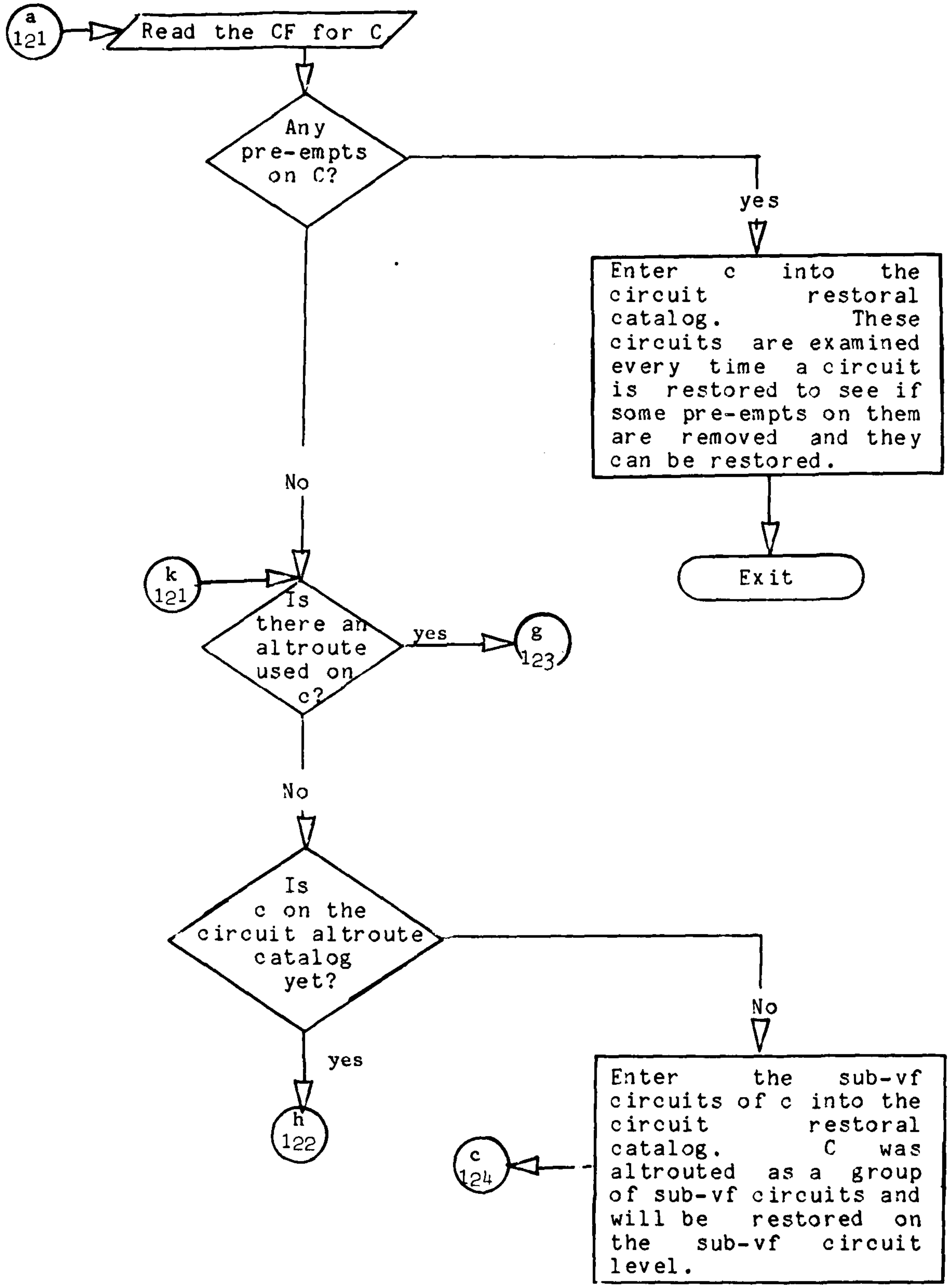
2.6.6.2 The Restoral Routine--

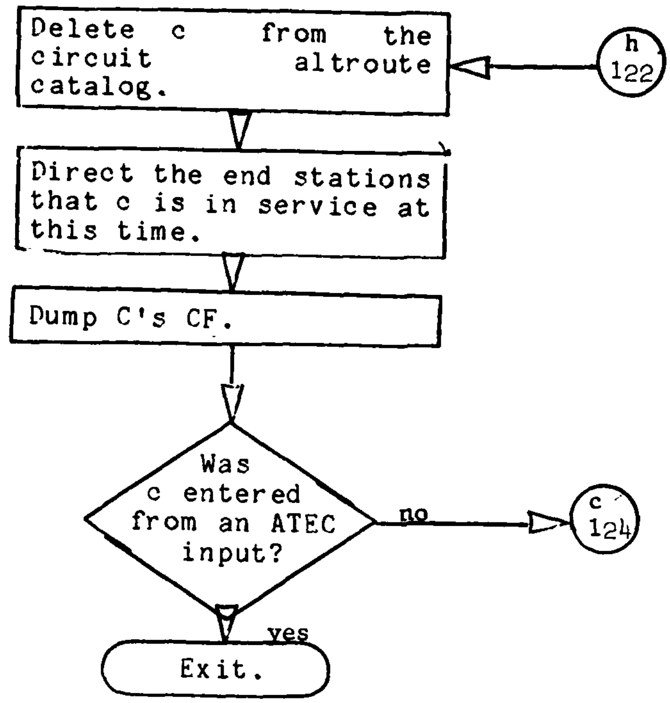



b 118
Is there an altroute used on t?
yes
e 119
no
Is t on the trunk altroute catalog yet?
No
Enter the circuits of t into the circuit restoral catalog. t was altrouted as a group of circuits and will be restored on the circuit level.
c 124
yes
Delete t from the trunk altroute catalog.
Direct the end stations that t is in service at this time.
Dump t's TF.
Was t entered from an ATEC input?
no
d 120
yes
Exit.

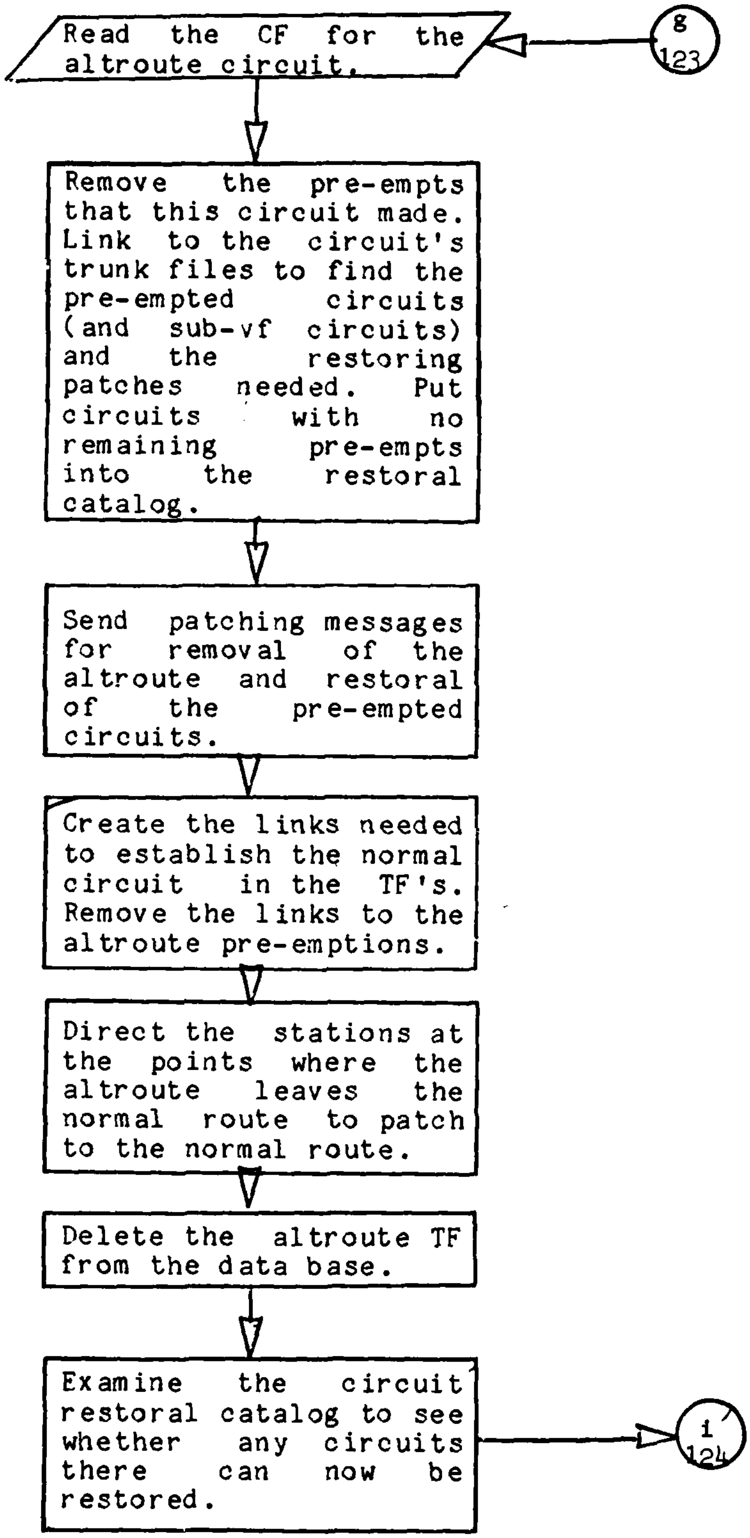

Read the TF for the altroute trunk.
e 119
Remove the pre-empts that this trunk made. Link to the trunk's link files to find the pre-empted trunks (and circuits on those trunks) and the restoring patches needed. Put circuits or trunks with no remaining pre-emptions in the restoral catalog.
Send patching messages for removal of the altroute and restoral of the pre-empted trunks.
Create the links needed to establish the normal trunk in the LF's. Remove the links to the altroute pre-emptions.
Direct the stations at the points where the altroute leaves the normal route to patch to the normal route.
Delete the altroute CF from the data base.
Examine the trunk restoral catalog to see whether any trunks or circuits there can now be restored.
f 120

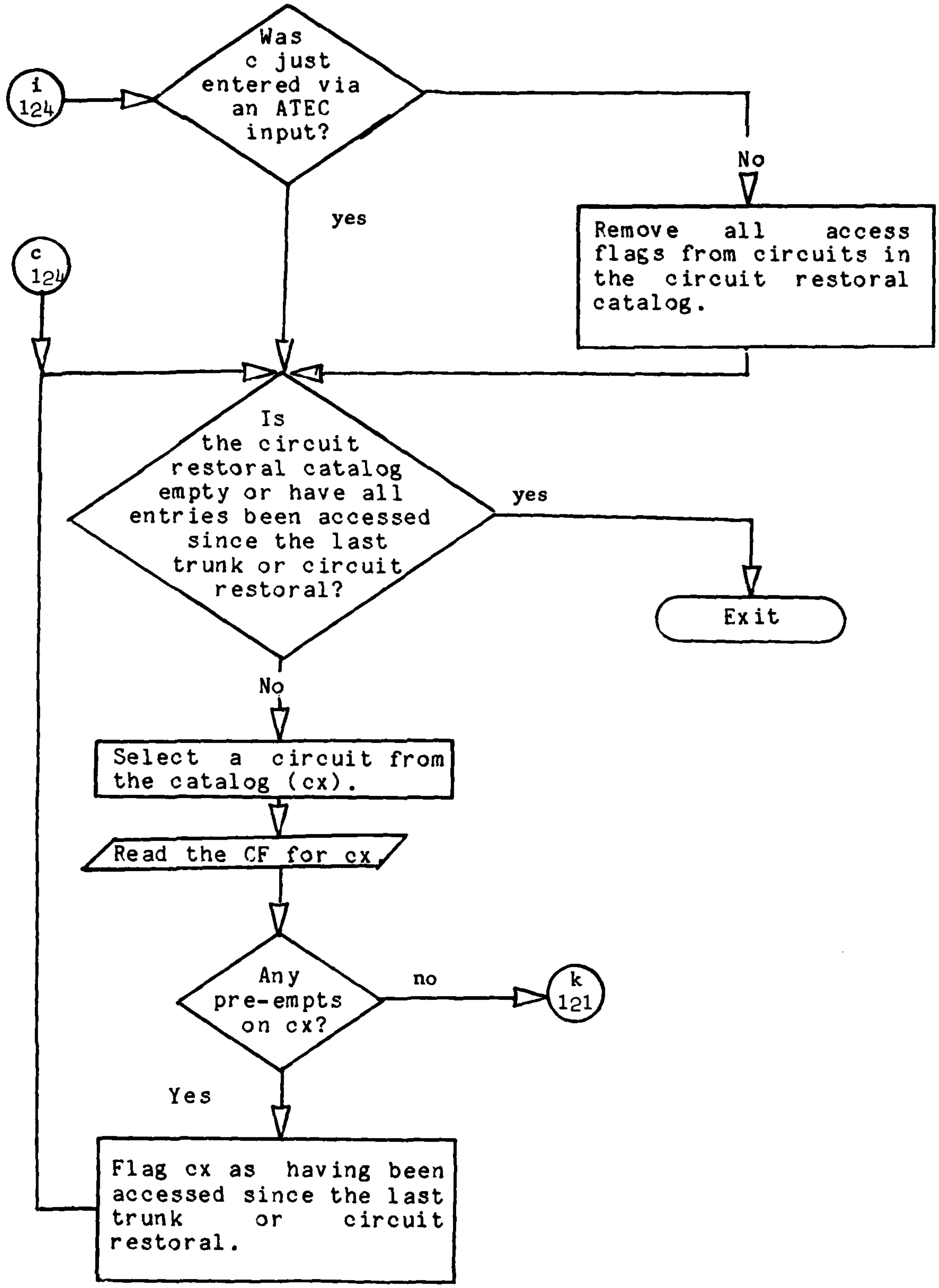

f 120
Was t just entered via an ATEC input?
no
yes
Remove all access flags from trunks and circuits in the restoral catalogs.
d 120
Is the trunk restoral catalog empty or have all entries been accessed since the last trunk restoral?
yes
c 124
No
Select a trunk from the catalog (tx).
Read the TF for tx.
Any pre-empts on tx?
no
b 118
Yes
Flag tx as having been accessed since the last trunk restoral.


a 121
Read the CF for C
Any pre-empts on C?
yes
Enter c into the circuit restoral catalog. These circuits are examined every time a circuit is restored to see if some pre-empts on them are removed and they can be restored.
Exit
No
k 121
Is there an altroute used on c?
yes
g 123
No
Is c on the circuit altroute catalog yet?
No
Enter the sub-vf circuits of c into the circuit restoral catalog. C was altrouted as a group of sub-vf circuits and will be restored on the sub-vf circuit level.
c 124
yes
h 122



Delete c from the circuit altroute catalog.
h
122
Direct the end stations that c is in service at this time.
Dump C's CF.
Was c entered from an ATEC input?
no
c
124
yes
Exit.


Read the CF for the altroute circuit.
g
123
Remove the pre-empts that this circuit made. Link to the circuit's trunk files to find the pre-empted circuits (and sub-vf circuits) and the restoring patches needed. Put circuits with no remaining pre-empts into the restoral catalog.
Send patching messages for removal of the altroute and restoral of the pre-empted circuits.
Create the links needed to establish the normal circuit in the TF's. Remove the links to the altroute pre-emptions.
Direct the stations at the points where the altroute leaves the normal route to patch to the normal route.
Delete the altroute TF from the data base.
Examine the circuit restoral catalog to see whether any circuits there can now be restored.
i
124


i 124
Was c just entered via an ATEC input?
No
yes
Remove all access flags from circuits in the circuit restoral catalog.
c 124
Is the circuit restoral catalog empty or have all entries been accessed since the last trunk or circuit restoral?
yes
Exit
No
Select a circuit from the catalog (cx).
Read the CF for cx.
Any pre-empts on cx?
no
k 121
Yes
Flag cx as having been accessed since the last trunk or circuit restoral.

2.7 ALTROUTING WITH AN AUTOMATED PATCHING NETWORK.

2.7.1 Changes in the Altrouting Concept

The presense of automated patching in the DCS network makes a significant difference in the structure of the service hierarchy and the way it is utilized in altrouting. The nodes of the network would have "channel reconfiguration units" (CRU's) installed to provide the circuit and group re-configuration at the station. These units would allow any channel of any group or link to be rerouted onto any other link, group or channel at the station without the multiplexer equipment present. Thus, each trunk now becomes exactly one link long. The concept of trunk and link are now one and the trunk can be deleted from use. The altrouting of groups in this case is also no longer needed because circuit group altrouting requires little more effort than altrouting the group intact. Also, altrouting groups of circuits is more efficient in that it concentrates the altrouting effort on the most important circuits before dealing with lesser circuits. Since we are not likely to be able to altroute entire groups anyway, limiting altrouting to circuits in this network is chosen.

The other major change in the altrouting routines due to the CRU network is the number of stations accessed in altroute expansion from a station. In the current trunk network, we are never guarenteed that an altroute can be found from the current search station by expanding to only the next nearest stations because the logical connectivity of the network and the trunk connectivity are not the same. We end up looking at all trunks available at a station so that all other accessible stations are examined at each search station. Even when an altroute is possible, we must look at all possible accessible stations in order to aviod "backhauling" or looping in the route found. In the CRU network, the most direct altroute can be found by expanding each search station to only its nearest neighbors before looking further. We are guarenteed that any altroute that can exist will be found by this link-by-link search. In the trunk network, we are never sure what link distance from a search station will yield the altroute. The result of this change in search policy is to reduce the disc access time for each station search and to eliminate "backhauling" and looping. The altroute found will be more direct and no doubt be less "costly" to implement.

Finally, a CRU network will reduce the time required to implement the altroutes due to its automated nature. This means that the grade of service as measured by circuit availability will improve. The CRU network will probably be limited in its altroute implementation time to the message transmission time involved to relay altroute patches to the CRU sites from the control center.

2.7.2 Data Base Modifications

The CRU network visualized here does not use the trunk concept - trunks and links are the same entity. Thus, all trunk files will be removed. To show the circuit content of the new one link trunks (i.e. links), the link files will need to be expanded in order to do the old trunk files' job. Table 2-1 shows the format of the new link file for the CRU network. The size of individual files increses dramatically due to the requirement that all carried circuits be visible in this file. There is a sizable data base increase with these new larger link files even though all trunk files are now deleted.

The other changes in data base files that occur deal with the files used during the altroute search routines. The TREE file need not list the link,group and channel for the parent station and new altroute station - they are the same due to the fact that the new altroute segment is only one link long. The field in TREE flagging the fact that a goal station was jumped over is no longer needed because that event cannot occur. The lists of altroute and restoral requirements are now only made for circuits because trunks and groups are not considerd as altroute hierarchy levels.

2.7.3 The Main Calling Routine

2.7.3.1 Discussion of Modifications to the Routine--The main calling routine changes in this case by deleting the trunk portion of the routine presented earlier. The failure of a link or group now results in the circuits carried being placed on the circuit altroute catalog directly without any attempt made to keep them intact as groups in altrouting.

Station no.	Parent station no.	Link no. of link leaving parent station (PS) for this station.	Group no. of group leaving PS for this station	Channel no. of channel leaving PS for this station	Cost to get from FS to this station via the PS. (COST)	Estimated cost to get to ITS from this station (HITS)	Estimated cost to get to TS from this station (HTS)	Search Status	Satellite channel moved as part of the link to the PS.
(1) 3 bytes	(2) 3 bytes	(3) 5 bytes	(4) 1 byte	(5) 2 bytes	(6)	(7)	(8)	(9) 1 byte	(10) 2 bytes

Satellite group the moved channel occupies	. . . (Data like the above fields for the other circuits in the re-route group being handled. Enter in the order of the circuit's RP's.)	(Data for last circuit in the altroute list.)	Satellite link the moved circuits occupy.	Connectivity path I. D. 5 for this station (if any). (up to 5 listed)
(11) 1 byte	(12)		(13) 5 bytes	(14) 10 bytes

FIGURE 2-10. THE STRUCTURE OF THE ENTRIES INTO FILE "TREE" FOR CIRCUIT ALTROUTING

TABLE 2-1. LINK FILE CONTENTS FOR CRU NETWORK

Item	Comments	Bytes
Link ID		5
Terminating Stations	Stations at each end of the link, for information and for alt route searching.	6
Circuits Carried. (Up to 384)	Group circuits by their carrying group. List nominal & current circuit. List channel, RP, and Port type. Use 4 byte CCSD's.	5792
DOD (Direction 1)	Degree of degradation (i.e., out or degraded)	1
Fault Pointer (Direction 1)	Points to first fault report, direction 1.	6
DOD (Direction 2)	Same as for Direction 1	1
Fault Pointer (Direction 2)	Same as for Direction 1	6
ETR and DTG	Estimated Time to Restore and Date/Time group when Estimate was made.	11
Highest RP	Highest restoration priority carried by the link to establish criticality of temporary/permanent restoral.	2
Connectivity Path ID		2
HAZCON		1
Data Base Distribution	List of all stations (2), nodes (2), sectors (2), and areas (2) to receive DB updates for this link. Use addressing as in ATEC 10000 Spec.	24
		5857

Number of Such Records = 410

Total Bytes = 410 x 5857 = 2,401,370.

TABLE 2-2 - CIRCUIT FILE CONTENTS

Item	Comments	Bytes
User	Identifies name of person to contact relative to this circuit.	12
Phone Number	Permits calling user.	10
RP	Restoration Priority used in impact analysis of outage.	2
VFCT Number	Identifies carrying trunk if this is a data circuit or the trunk record if this is a VFCT.	8
Link, group and Channel Number	For each segment and terminating station - up to 18.	126
Reroute ID #1 and Flag	Identifies the circuit which is pre-planned for restoral of this circuit, whether it is activated and whether it has failed and activated.	9
Reroute ID #2 and Flag	Identifies either a circuit (4 byte CCSD) other than the preplanned reroute which was used to restore this circuit or other circuits (5 max.) which have pre-empted this circuit. A flag indicates that this field is idle, or that this circuit has been rerouted, preempted or that the reroute has failed.	21
Degree of Degradation, Direction 1, and Fault Location	Identifies whether there is a complete outage or a degradation, and where the fault is. Direction 1 for circuit level faults.	4
Degree of Degradation, Direction 2, and Fault Location	Identifies whether there is a complete outage or a degradation, and where the fault is. Direction 2 for circuit level faults.	4
Port type	Identifies the type of port this circuit uses on a first-level mux..	1

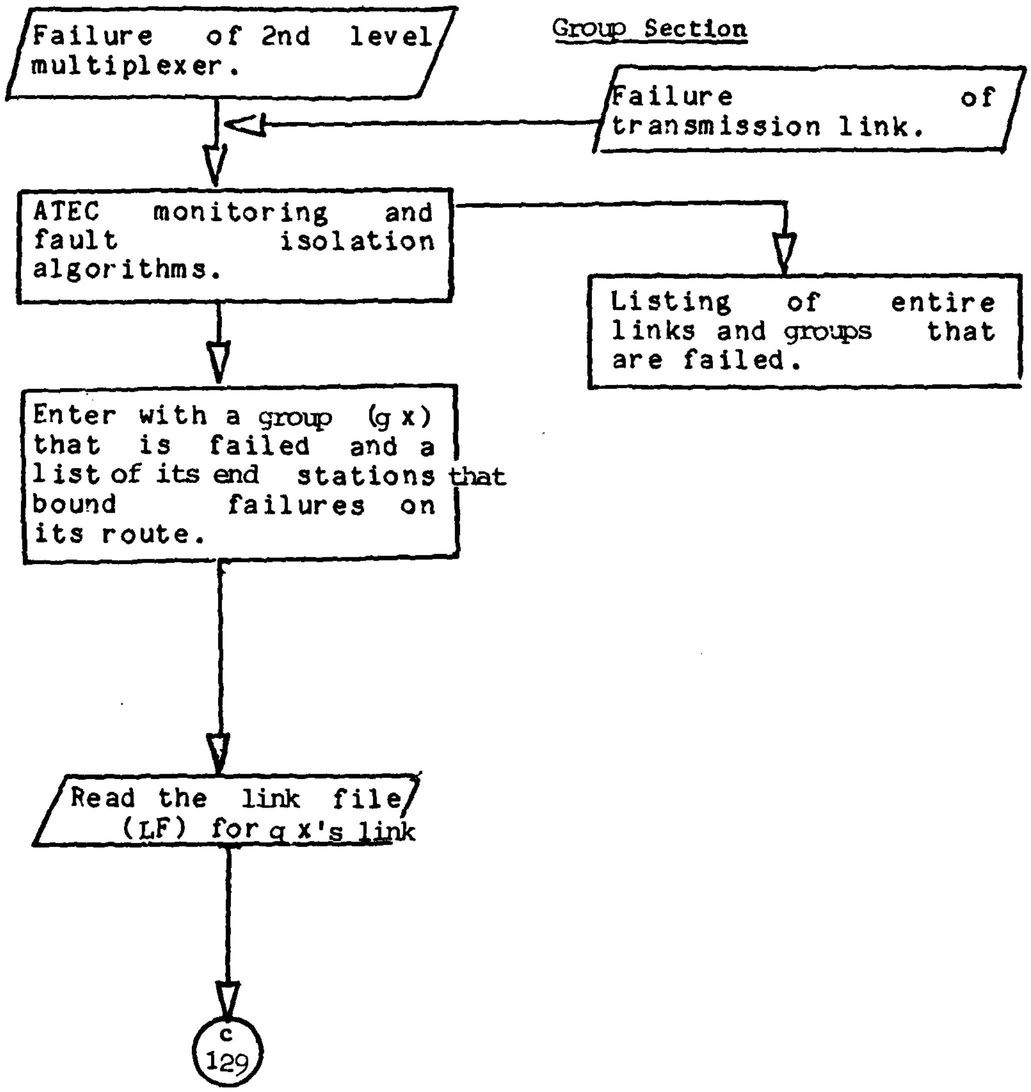
TABLE 2-2. CIRCUIT FILE CONTENTS (Continued)

Item	Comments	Bytes
Fault Pointer, Direction 1	Points to first fault report for direction 1.	6
Fault Pointer, Direction 2	Points to first fault report for direction 2.	6
Data Base Distribution	List all stations (6 x 3), nodes (3 x 4), sectors (3 x 4), and areas (2 x 3) to receive DB updates. Use addressing as in ATEC 10000 Spec.	48
Control Responsibility		3
Number of Such Records = 10,500*		260

Total Bytes = 10,500 x 260 = 2,730,000

*Based on 7,400 circuits in unclassified portion of 1978 DCS connectivity data base, intra and inter Europe. This was taken to be 90% of total circuits. A 25% growth factor was added.

2.7.3.2.
Modifications of the Main Calling Routine



c
129

Enter the circuits carried by the group into the circuit altroute catalog by RP ordering. List the stations bounding failures on the group with each entry as stations bounding the circuit failures.

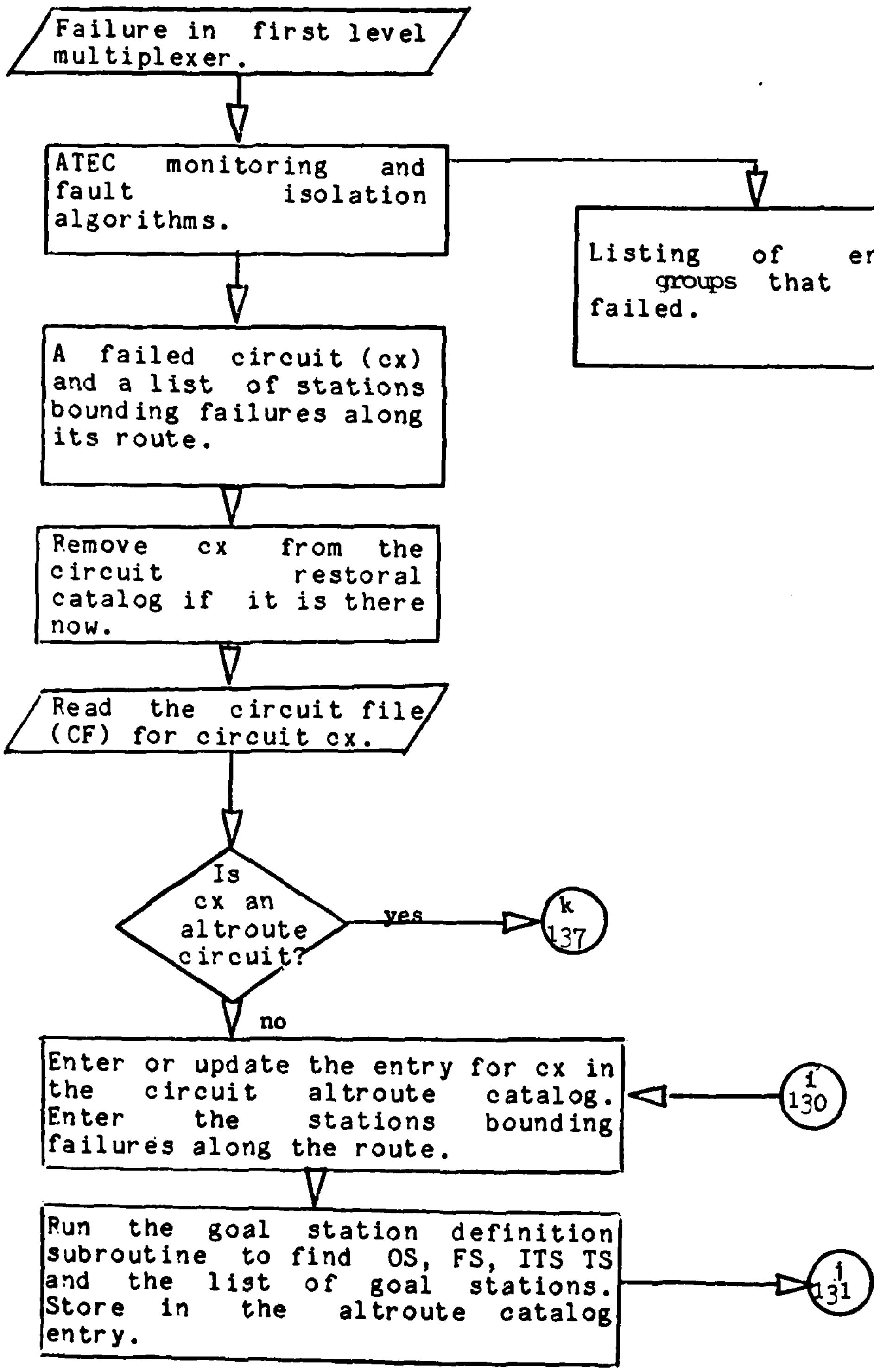
Transfer any pre-emptions of the trunk to pre-emptions in the CF's of the carried circuits.

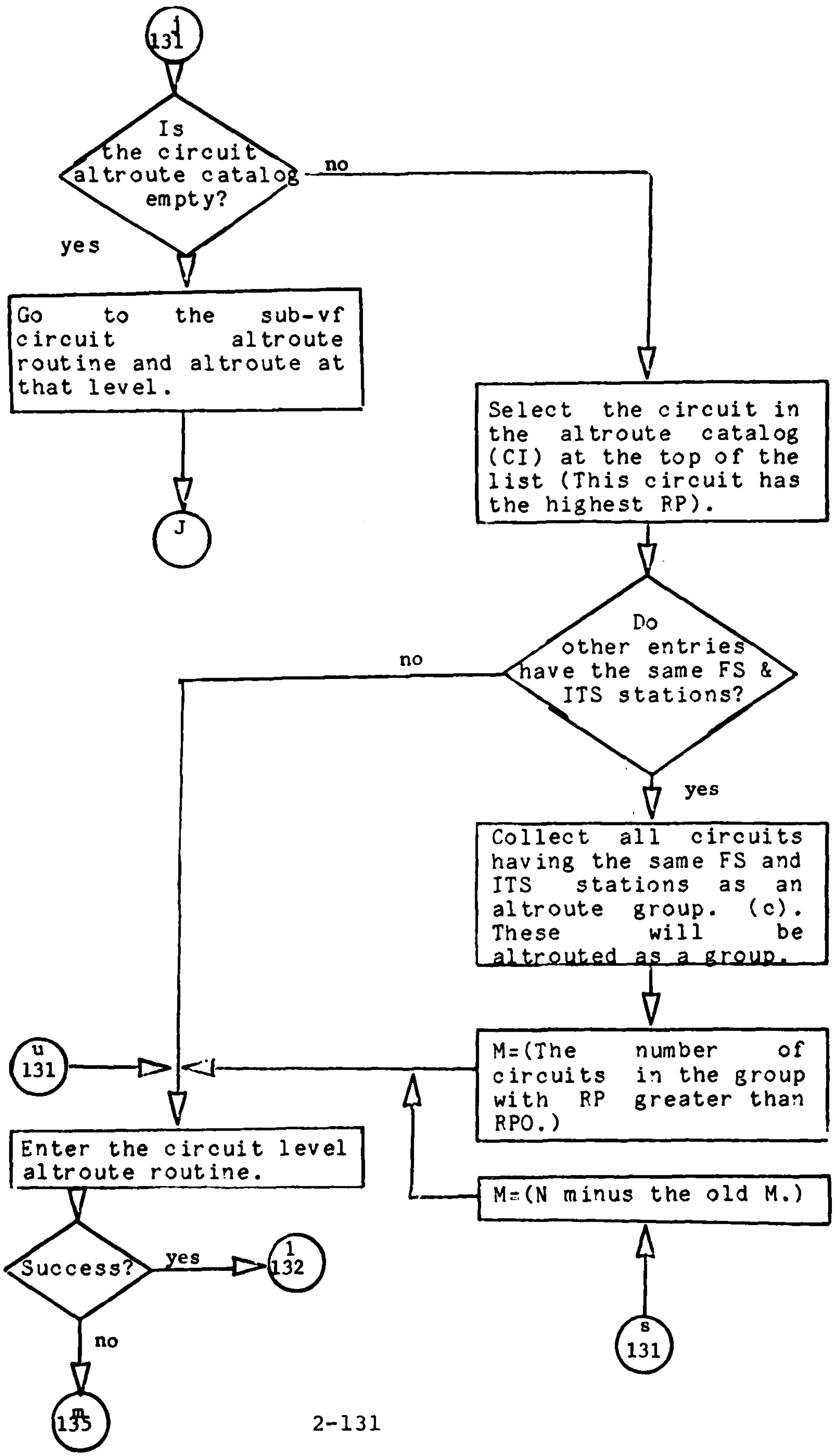
Run the circuit goal station definition routine for each circuit on the group and enter the results into a circuit altroute catalog entry for each circuit.

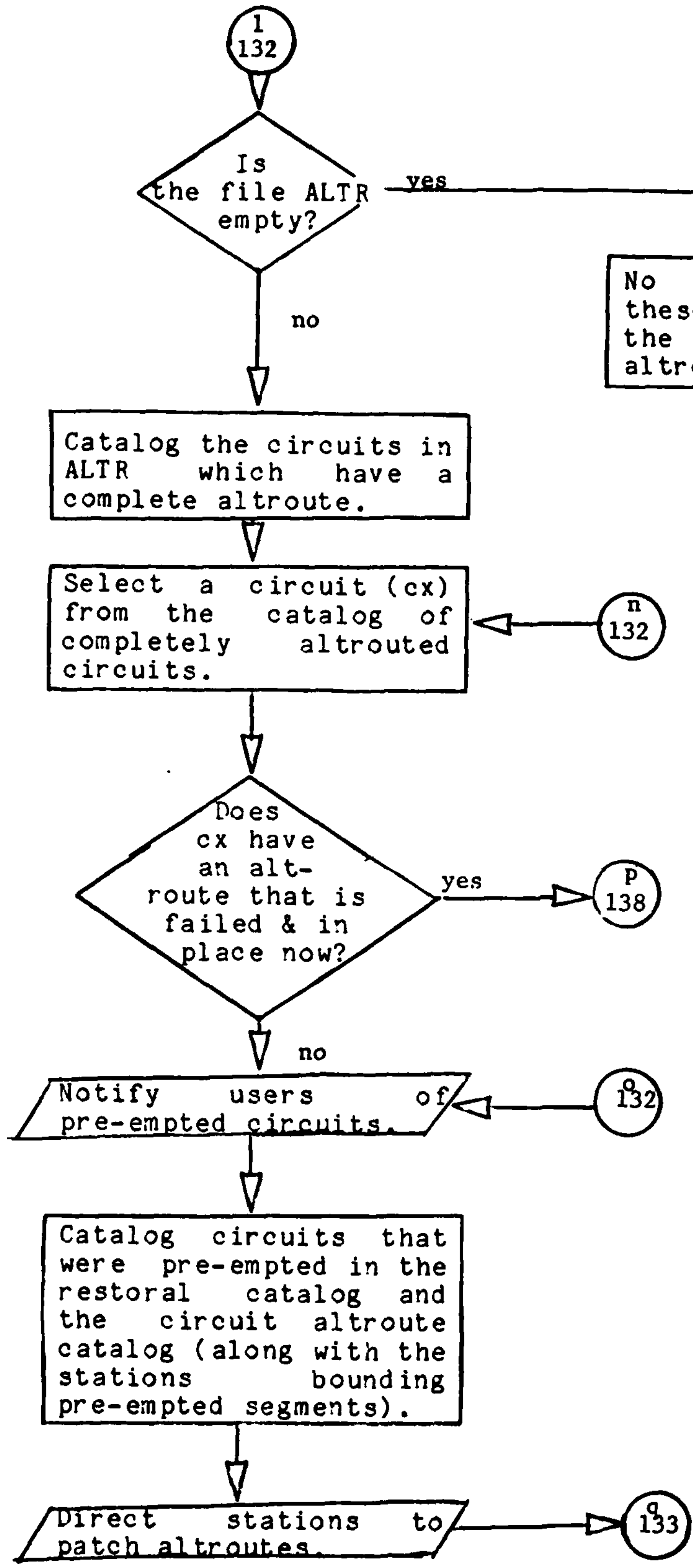
Do not keep a circuit altroute entry for circuits which fail the goal station definition search. They are isolated on a spur and cannot be altrouted.

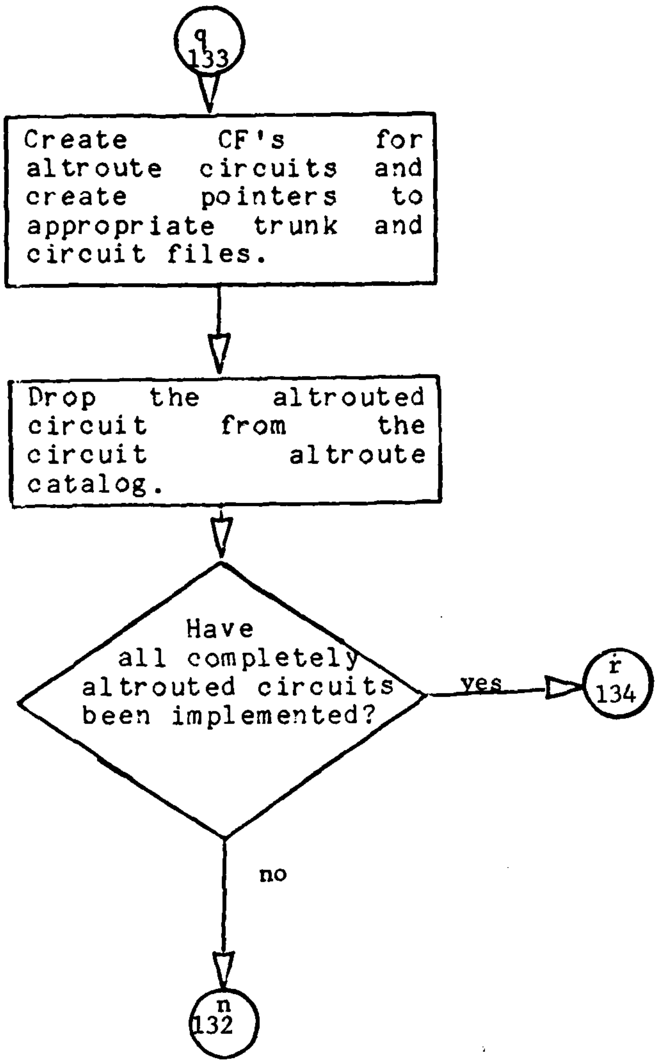
j
131

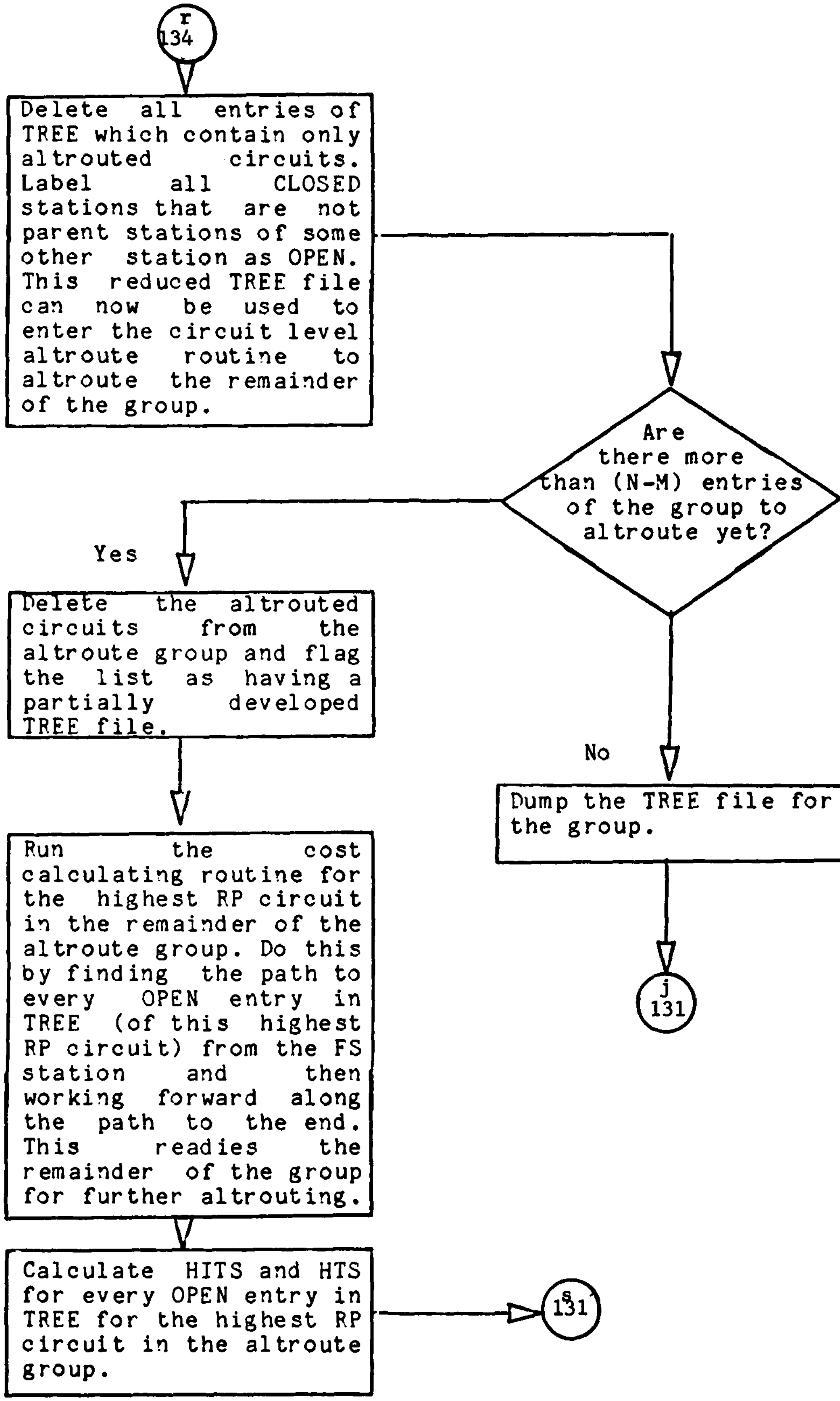
Circuit Section

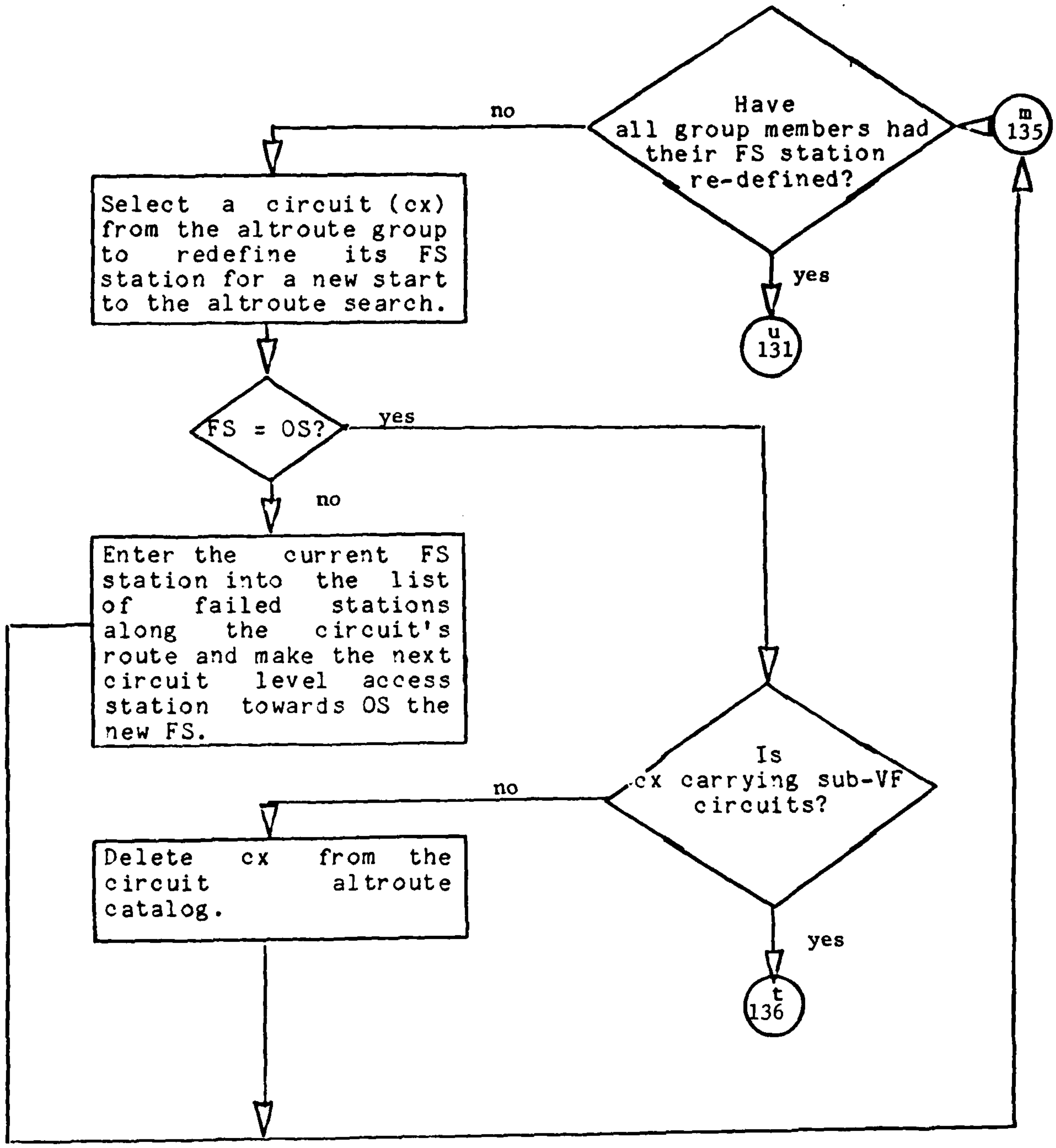



j
131
Is the circuit altroute catalog empty?
no
yes
Go to the sub-vf circuit altroute routine and altroute at that level.
J
Select the circuit in the altroute catalog (CI) at the top of the list (This circuit has the highest RP).
Do other entries have the same FS & ITS stations?
no
yes
Collect all circuits having the same FS and ITS stations as an altroute group. (c). These will be altrouted as a group.
u
131
M=(The number of circuits in the group with RP greater than RPO.)
Enter the circuit level altroute routine.
M=(N minus the old M.)
Success?
yes
1
132
no
s
131
m
135


l
132
Is the file ALTR empty?
yes
no
No need to altroute these circuits. Drop the list from the altroute catalog.
j
131
Catalog the circuits in ALTR which have a complete altroute.
Select a circuit (cx) from the catalog of completely altrouted circuits.
n
132
Does cx have an alt-route that is failed & in place now?
yes
p
138
no
Notify users of pre-empted circuits.
o
132
Catalog circuits that were pre-empted in the restoral catalog and the circuit altroute catalog (along with the stations bounding pre-empted segments).
Direct stations to patch altroutes.
q
133


q
133
Create CF's for altroute circuits and create pointers to appropriate trunk and circuit files.
Drop the altrouted circuit from the circuit altroute catalog.
Have all completely altrouted circuits been implemented?
yes
r
134
no
n
132


r
134
Delete all entries of TREE which contain only altrouted circuits. Label all CLOSED stations that are not parent stations of some other station as OPEN. This reduced TREE file can now be used to enter the circuit level altroute routine to altroute the remainder of the group.
Are there more than (N-M) entries of the group to altroute yet?
Yes
Delete the altrouted circuits from the altroute group and flag the list as having a partially developed TREE file.
No
Dump the TREE file for the group.
j
131
Run the cost calculating routine for the highest RP circuit in the remainder of the altroute group. Do this by finding the path to every OPEN entry in TREE (of this highest RP circuit) from the FS station and then working forward along the path to the end. This readies the remainder of the group for further altrouting.
Calculate HITS and HTS for every OPEN entry in TREE for the highest RP circuit in the altroute group.
s
131


Have all group members had their FS station re-defined?
no
yes
m
135
u
131
Select a circuit (cx) from the altroute group to redefine its FS station for a new start to the altroute search.
FS = OS?
yes
no
Enter the current FS station into the list of failed stations along the circuit's route and make the next circuit level access station towards OS the new FS.
Is cx carrying sub-VF circuits?
no
yes
Delete cx from the altroute circuit catalog.
t
136

t
136

Enter the circuits into the catalog of sub-vf circuits to altroute. Enter the sub-vf circuits carried by the vf circuit into the catalog of sub-vf circuits to altroute. Order them by RP. List the station ends of the carrying circuit as stations bounding the failures on the sub-vf circuit.

Place the VF circuits and sub-vf circuits on the circuit restoral catalog.

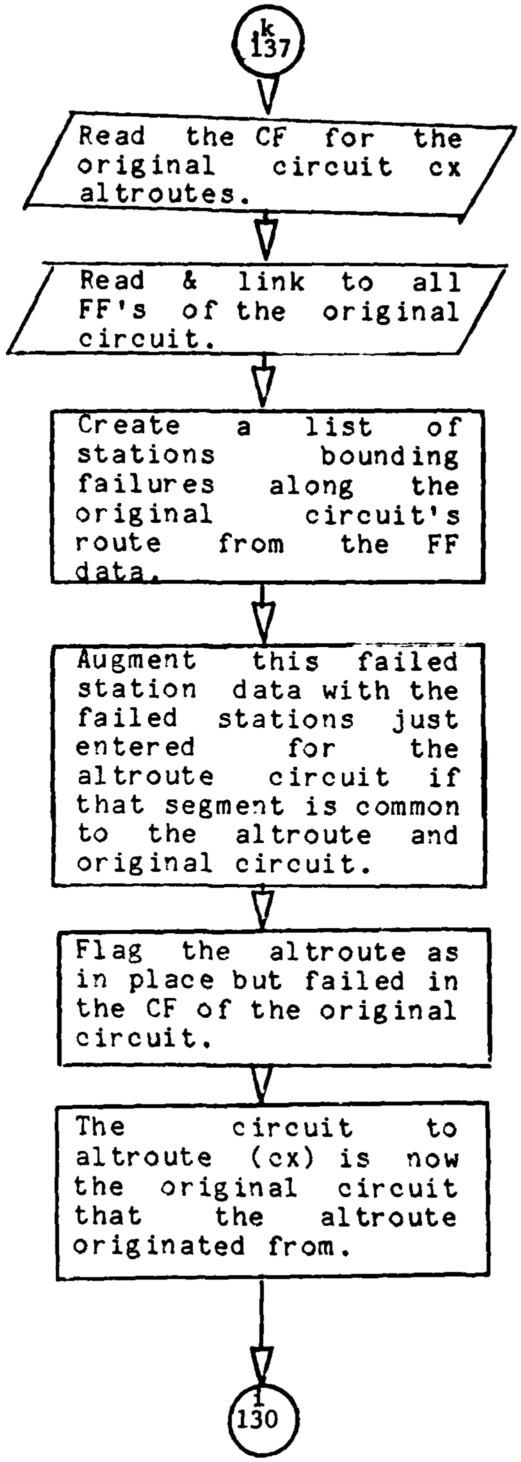
Carry over any pre-emptions on the vf circuit to the sub-vf circuits being carried.

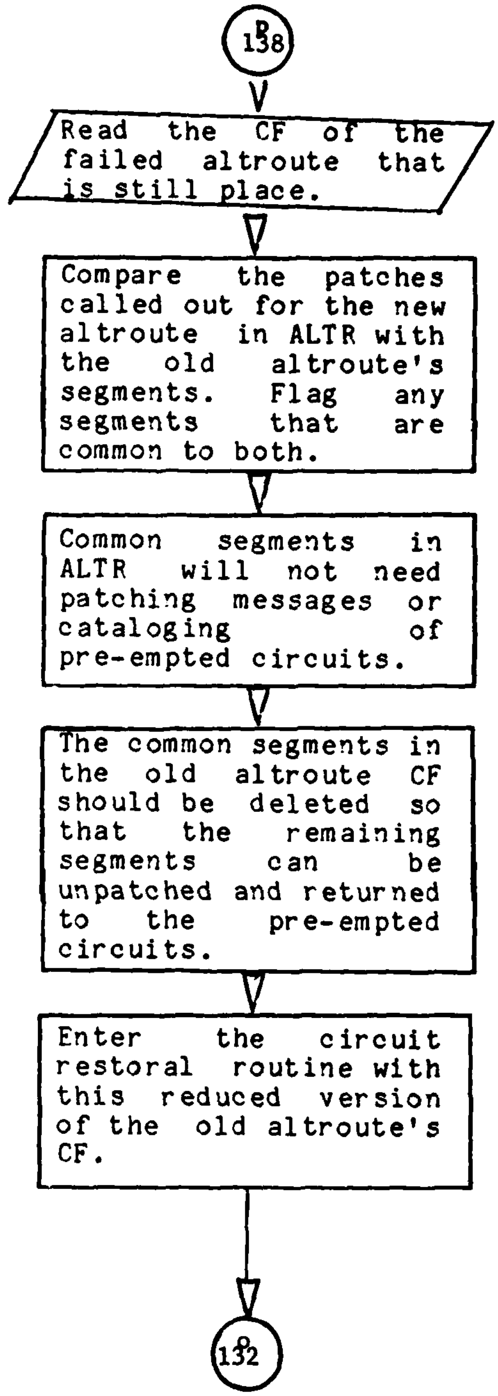
Run the sub-vf circuit goal station definition routine to find the OS, FS, ITS, TS and goal stations of the circuit.

Remove cx from the circuit altroute and restoral catalogs.

Do not keep a circuit which leaves the goal station definition routine in failure. That circuit is isolated on a spur and cannot be altrouted.

m
135

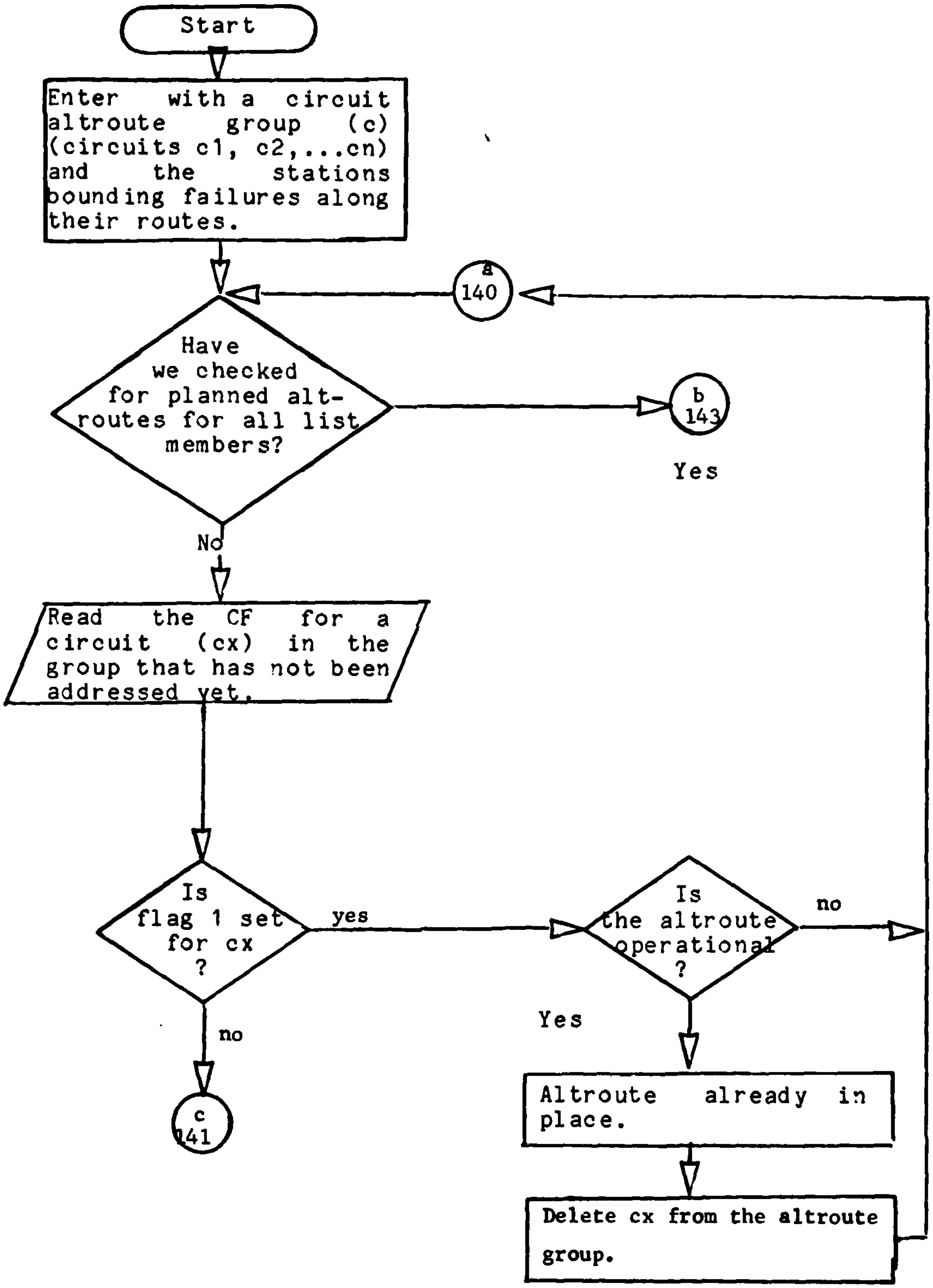

k 137
Read the CF for the original circuit cx altroutes.
Read & link to all FF's of the original circuit.
Create a list of stations bounding failures along the original circuit's route from the FF data.
Augment this failed station data with the failed stations just entered for the altroute circuit if that segment is common to the altroute and original circuit.
Flag the altroute as in place but failed in the CF of the original circuit.
The circuit to altroute (cx) is now the original circuit that the altroute originated from.
l 130

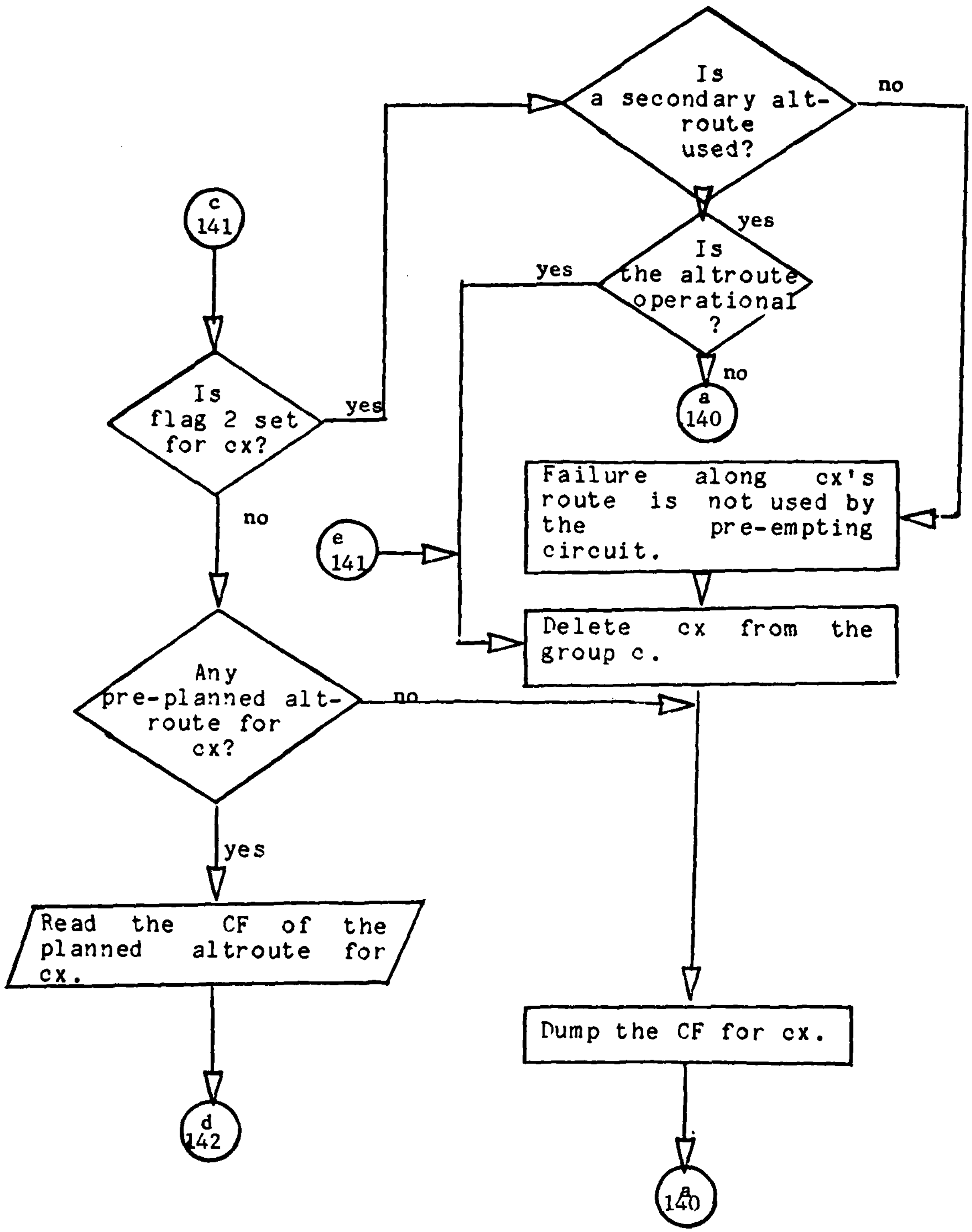

P 138
Read the CF of the failed altroute that is still place.
Compare the patches called out for the new altroute in ALTR with the old altroute's segments. Flag any segments that are common to both.
Common segments in ALTR will not need patching messages or cataloging of pre-empted circuits.
The common segments in the old altroute CF should be deleted so that the remaining segments can be unpatched and returned to the pre-empted circuits.
Enter the circuit restoral routine with this reduced version of the old altroute's CF.
P 132

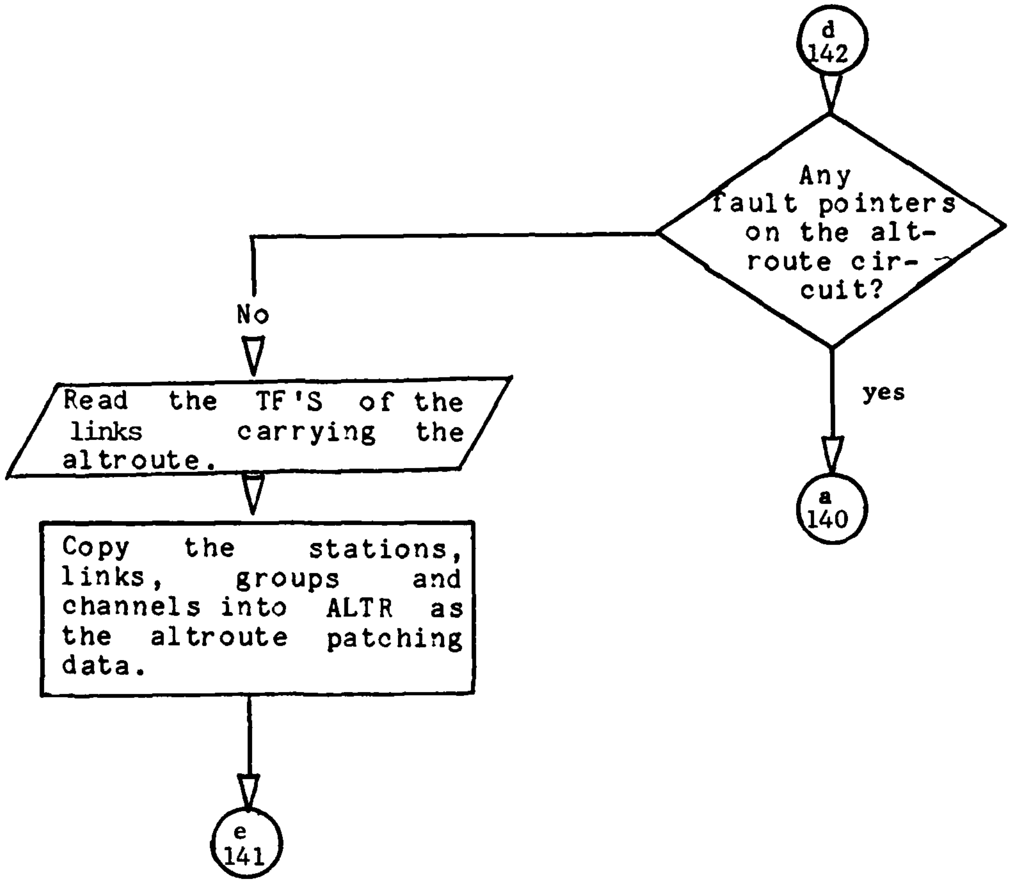
2.7.4 The Circuit Altrouting Routine

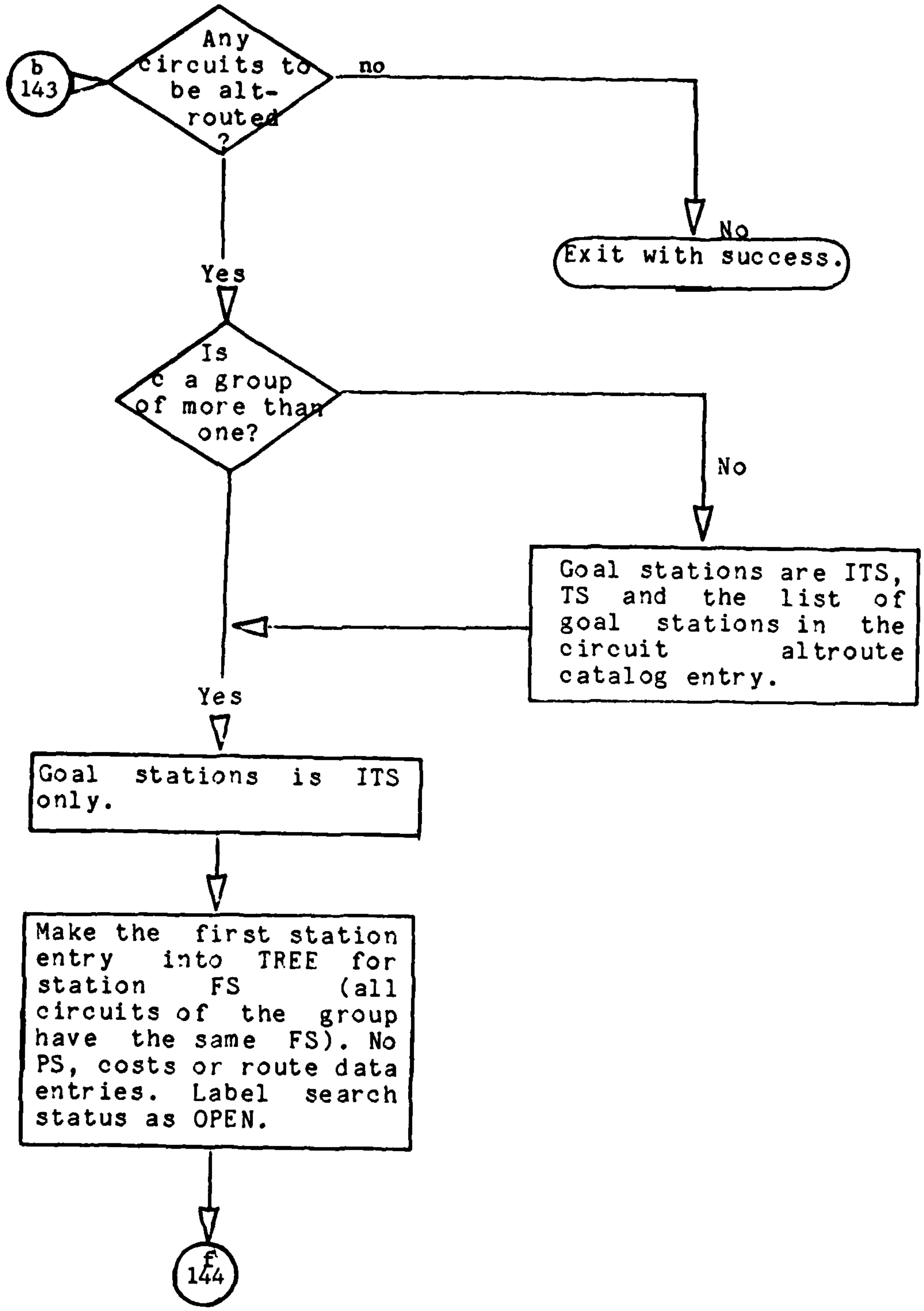
2.7.4.1 Discussion of the Modifications--The circuit altrouting routine has several major simplifications due to the CRU network. The search for pre-emptable circuits is made only for the station on the other end of the links connected to the station. The reading of the link file gives the routine all of the data it needs to assign pre-emptable circuits to the altroute group. No trunk files are needed. There is also no need to worry about a "backhaul" in the expansion of a station. Finally, the need to collect partial altroute circuit groups to the expanded station is not necassary - all circuits to the station are examined at once from the link file to that station.

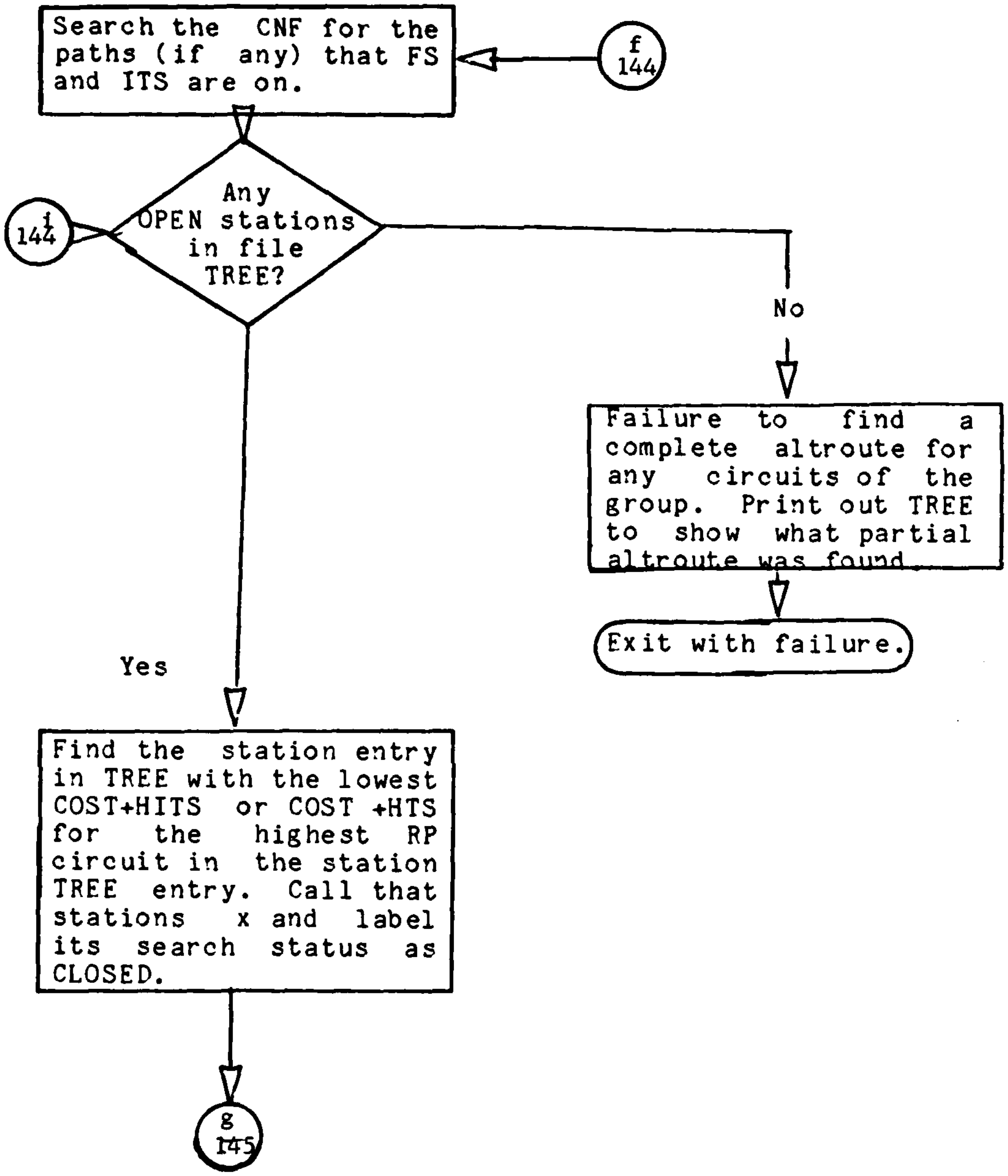
2.7.4.2.
Modifications of the Circuit Altrouting Routine

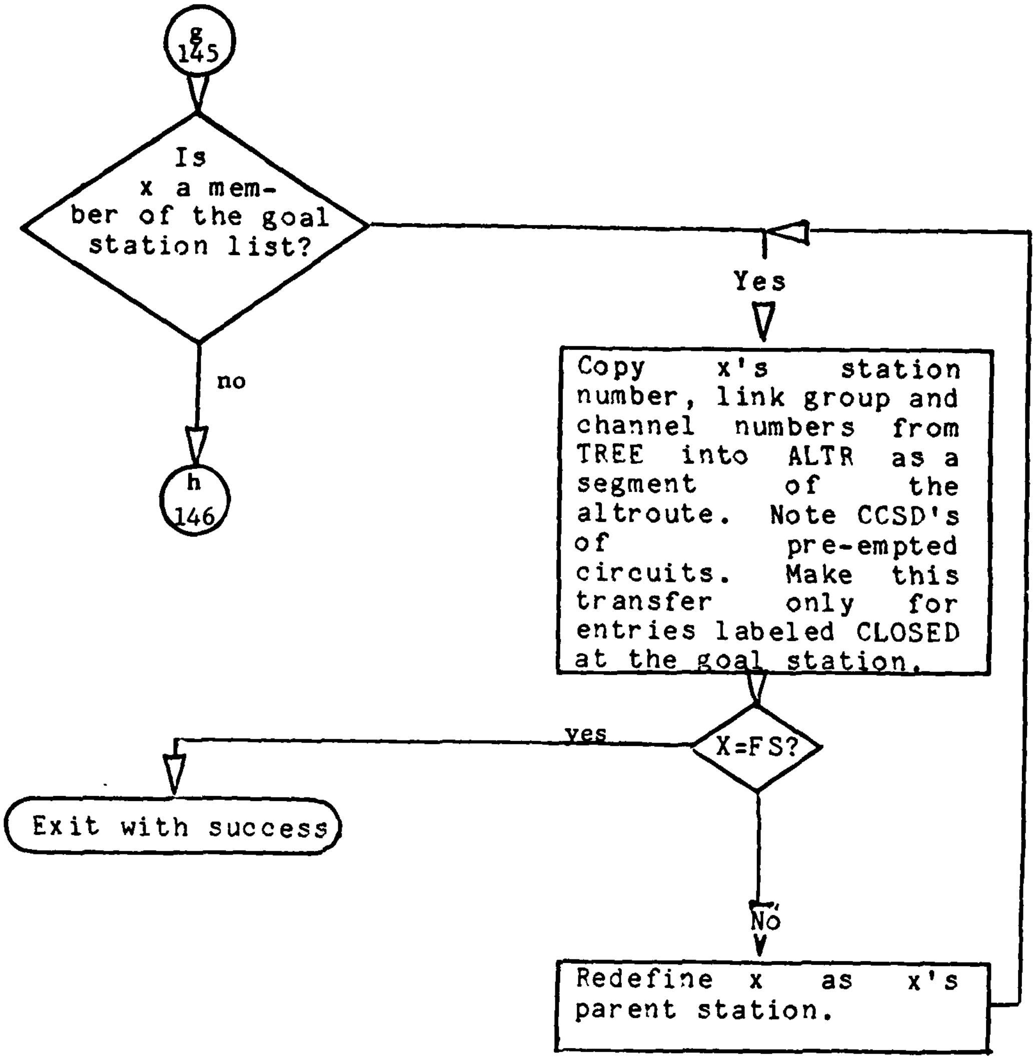


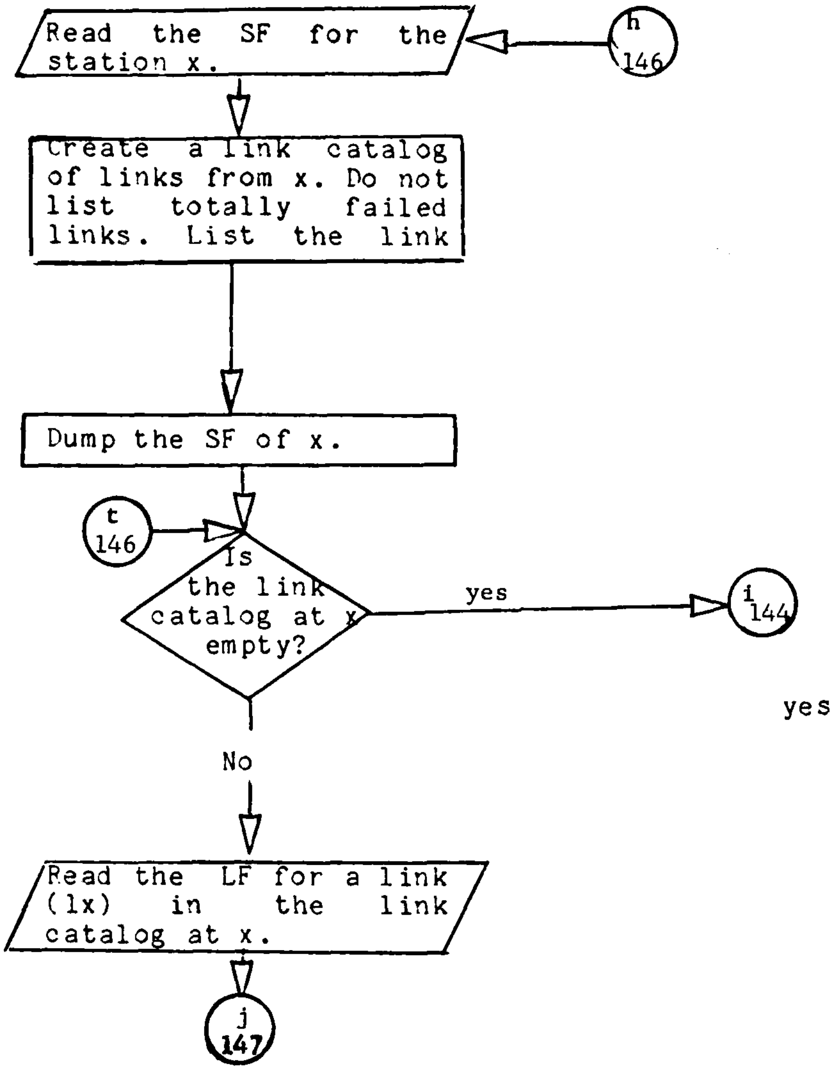

c 141
Is flag 2 set for cx?
yes
no
Is a secondary alt-route used?
no
yes
Is the altroute operational ?
yes
no
a 140
Failure along cx's route is not used by the pre-empting circuit.
e 141
Delete cx from the group c.
Any pre-planned alt-route for cx?
no
yes
Read the CF of the planned altroute for cx.
d 142
Dump the CF for cx.
a 140

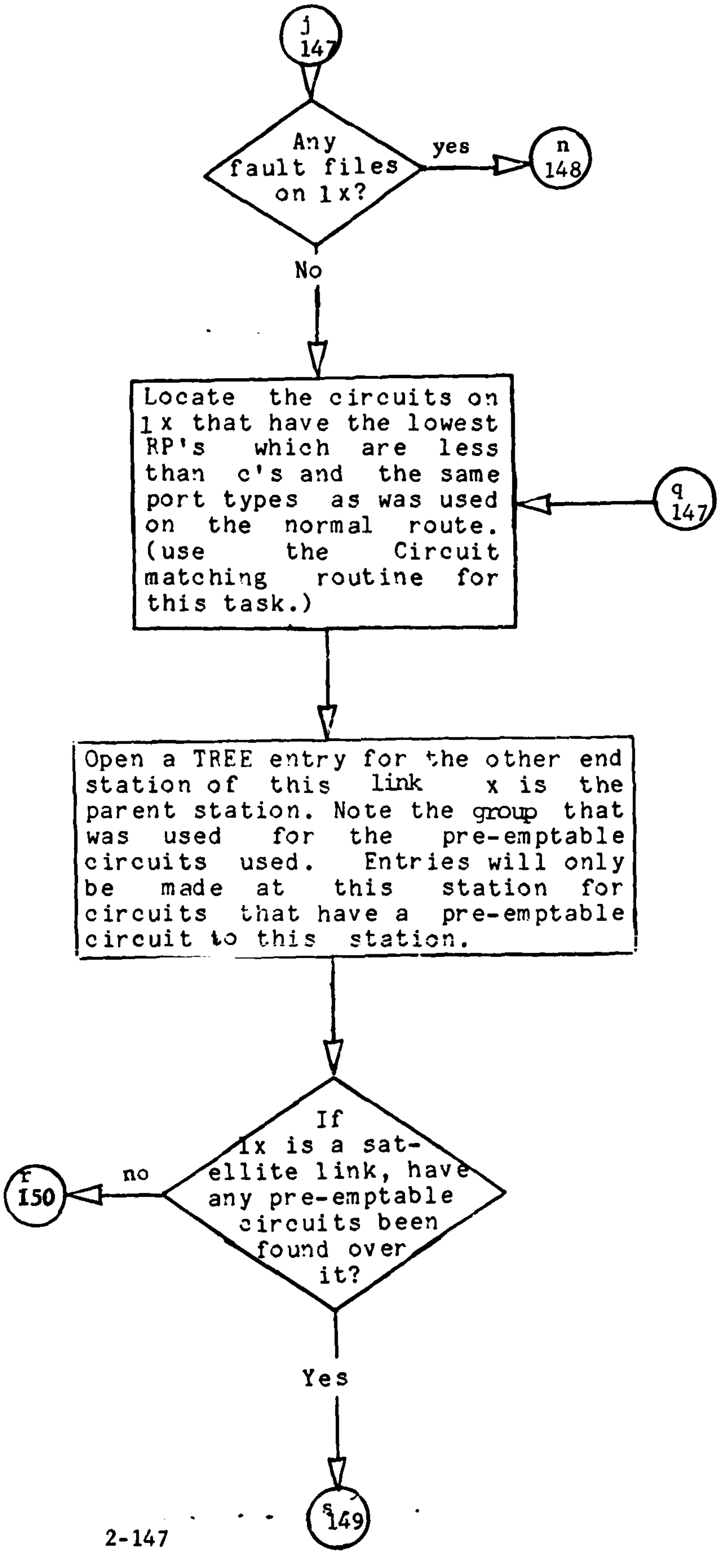

d
142
Any fault pointers on the alt-route cir-cuit?
No
yes
Read the TF'S of the links carrying the altroute.
Copy the stations, links, groups and channels into ALTR as the altroute patching data.
e
141
a
140

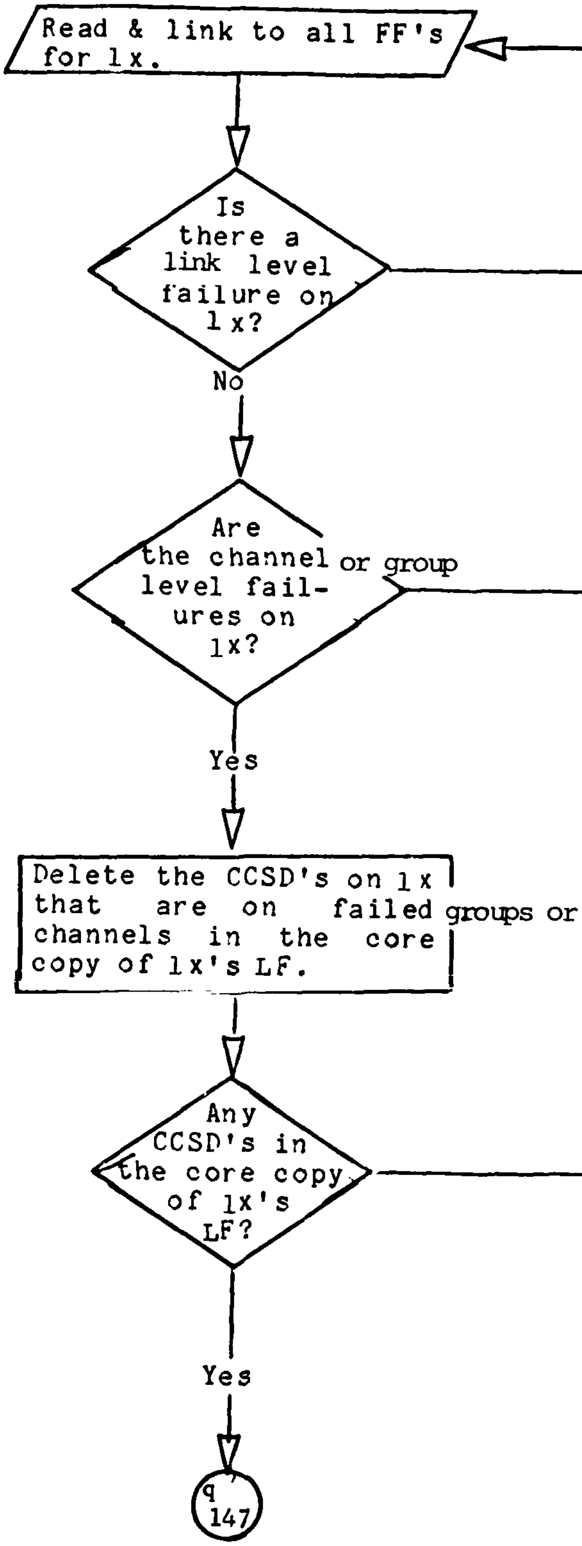

b 143
Any circuits to be alt-routed ?
no
Yes
No
Exit with success.
Is c a group of more than one?
No
Goal stations are ITS, TS and the list of goal stations in the circuit altroute catalog entry.
Yes
Goal stations is ITS only.
Make the first station entry into TREE for station FS (all circuits of the group have the same FS). No PS, costs or route data entries. Label search status as OPEN.
f 144

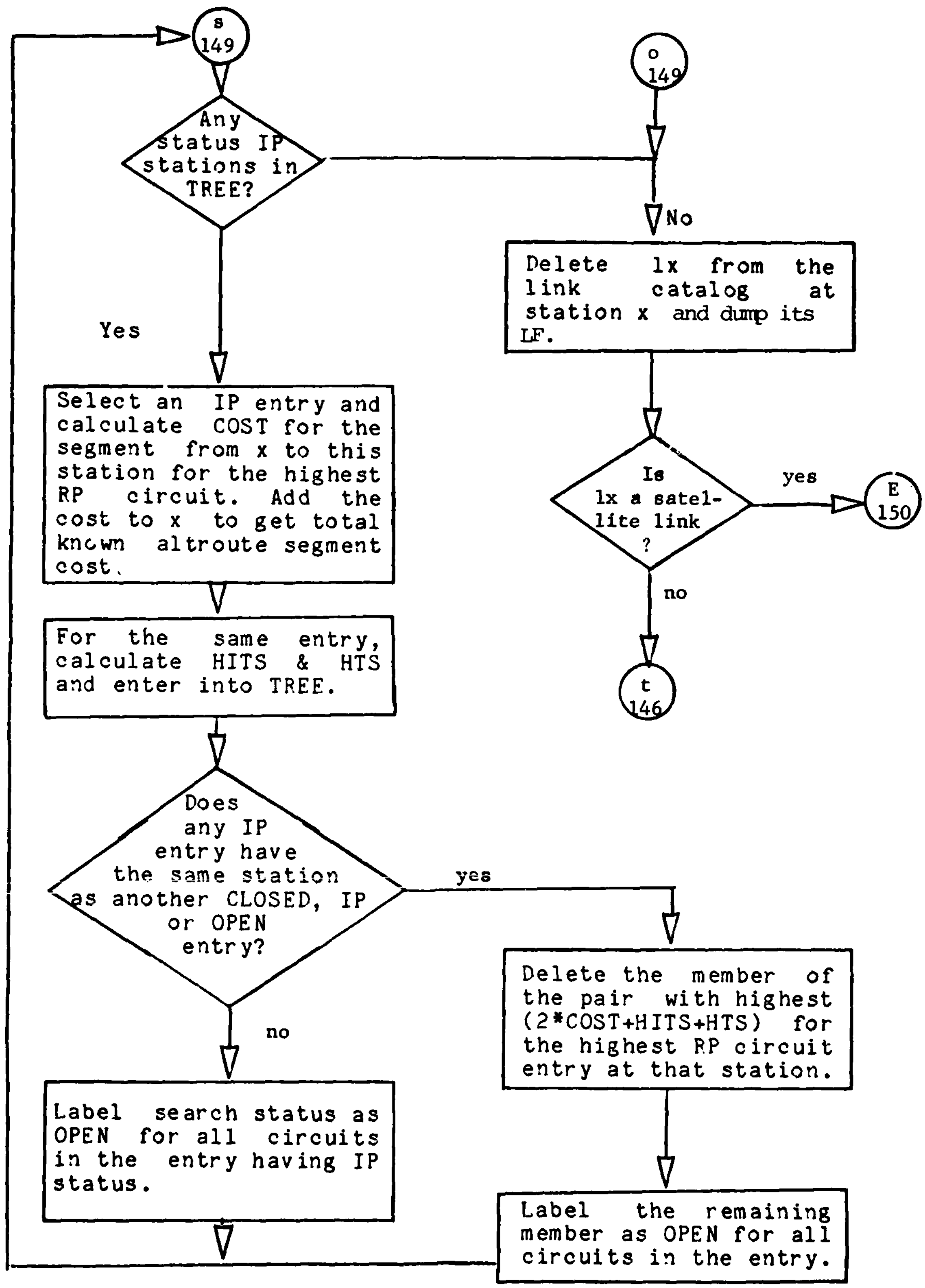

Search the CNF for the paths (if any) that FS and ITS are on.
f 144
Any OPEN stations in file TREE?
i 144
No
Failure to find a complete altroute for any circuits of the group. Print out TREE to show what partial altroute was found.
Exit with failure.
Yes
Find the station entry in TREE with the lowest COST+HITS or COST +HTS for the highest RP circuit in the station TREE entry. Call that stations x and label its search status as CLOSED.
g 145

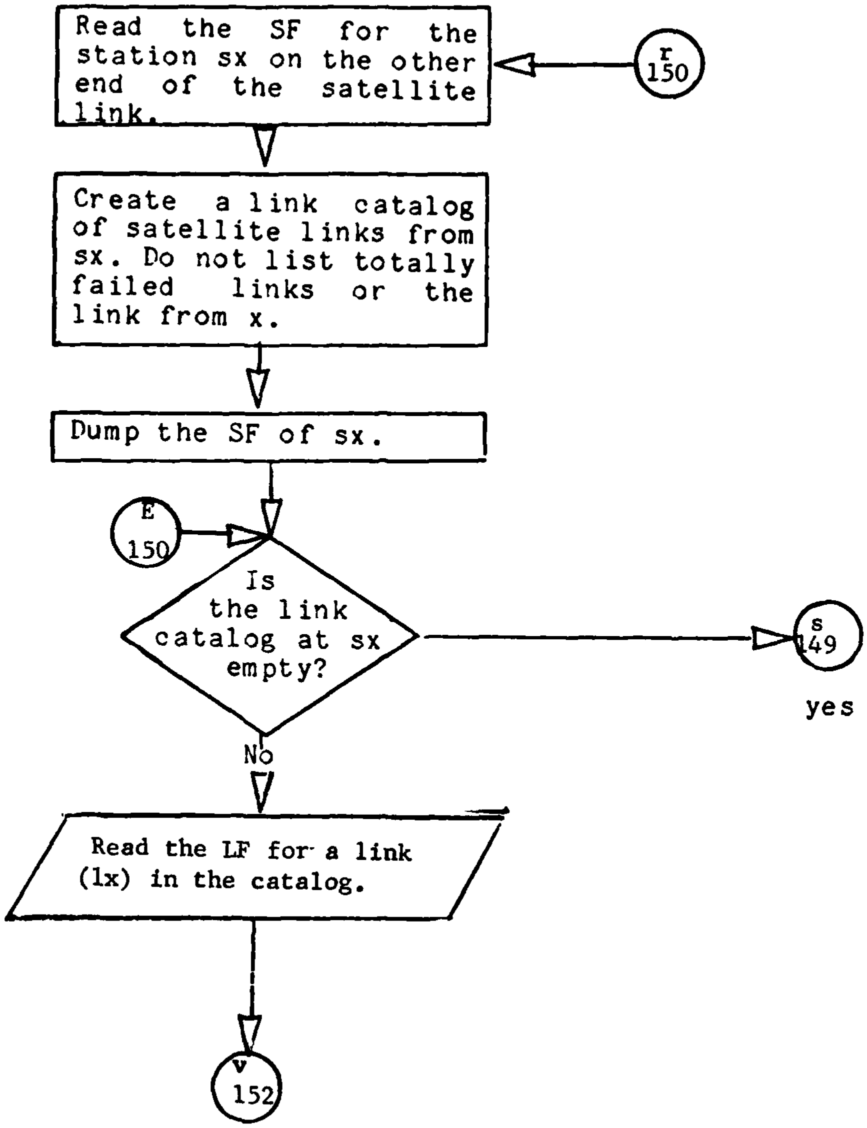

g
145
Is x a member of the goal station list?
no
h
146
Yes
Copy x's station number, link group and channel numbers from TREE into ALTR as a segment of the altroute. Note CCSD's of pre-empted circuits. Make this transfer only for entries labeled CLOSED at the goal station.
X=FS?
yes
Exit with success
No
Redefine x as x's parent station.

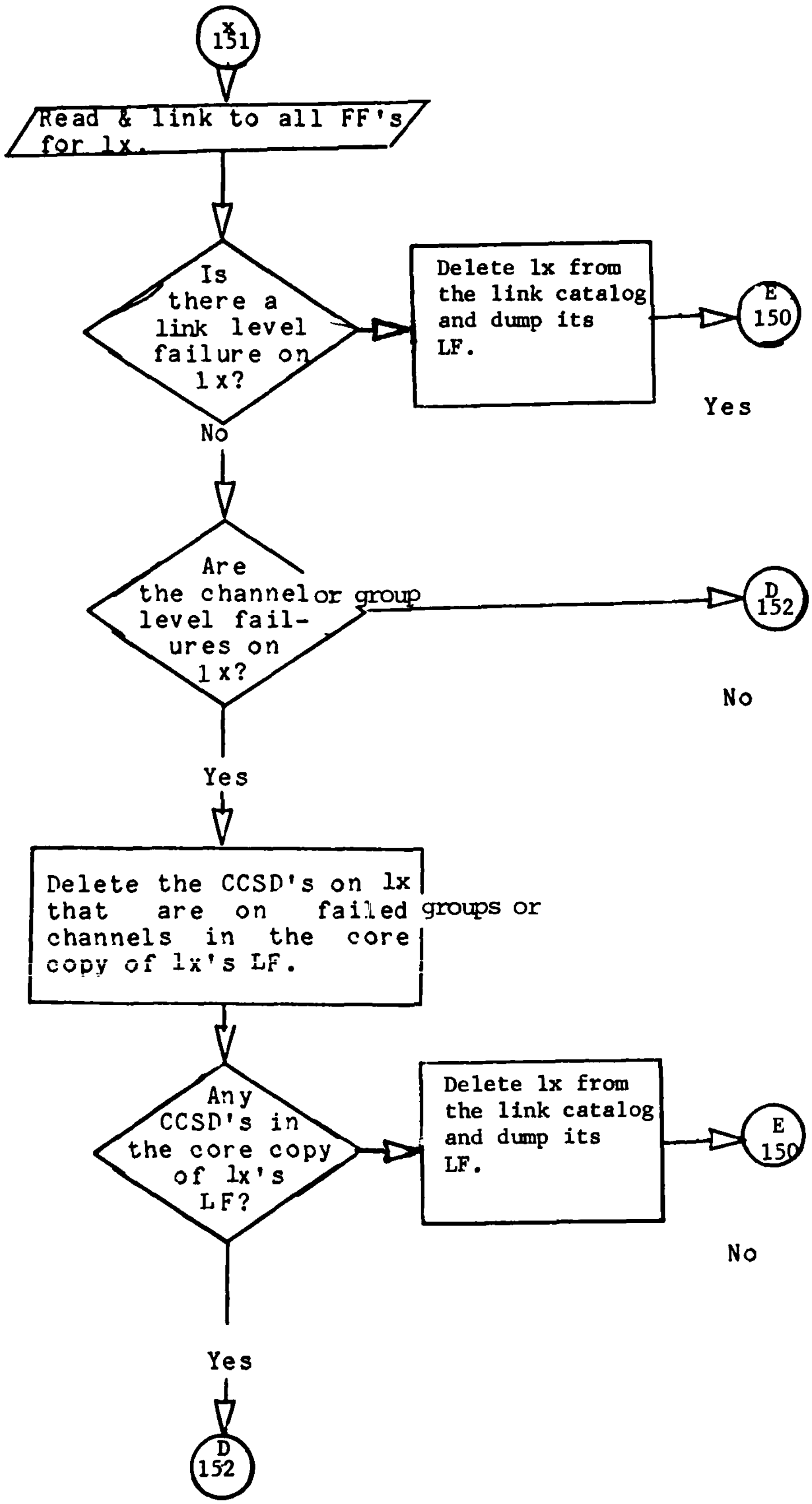

Read the SF for the station x.
h 146
Create a link catalog of links from x. Do not list totally failed links. List the link
Dump the SF of x.
t 146
Is the link catalog at x empty?
yes
i 144
yes
No
Read the LF for a link (lx) in the link catalog at x.
j 147

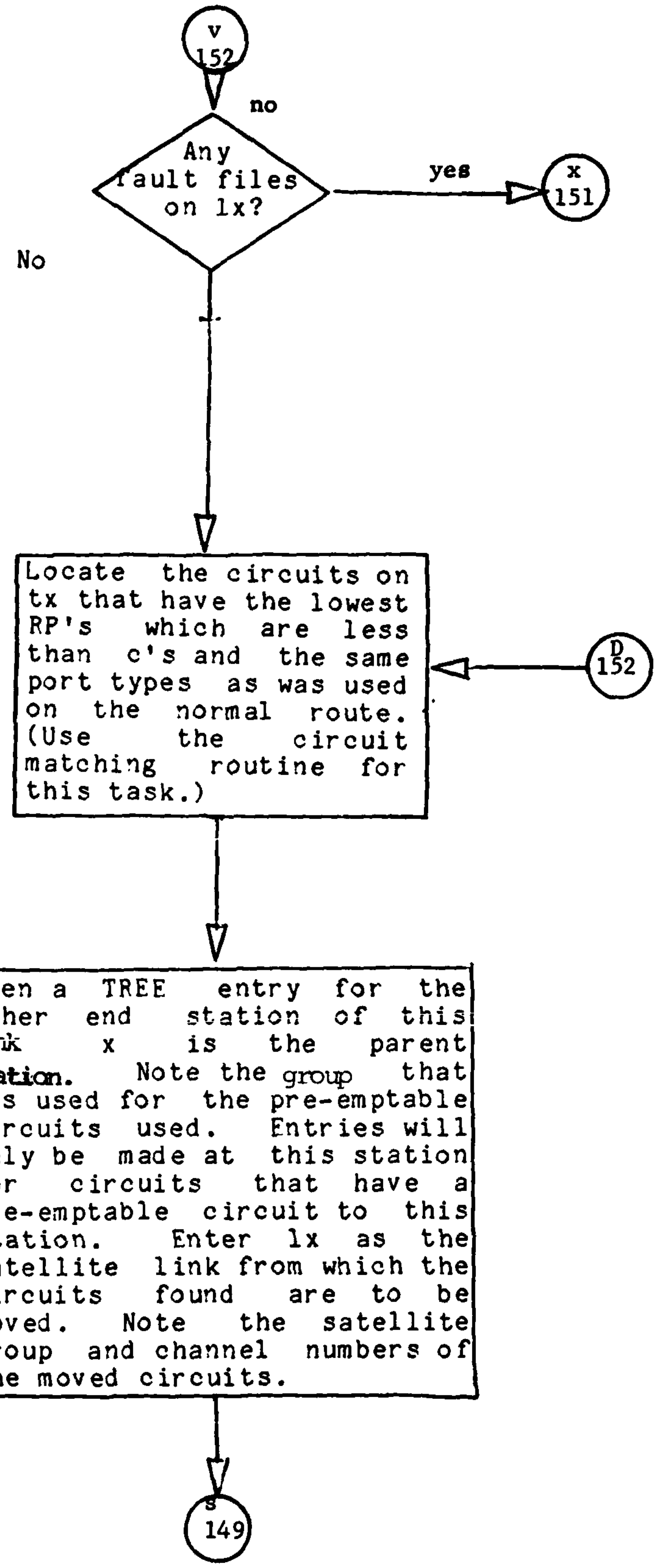

j 147
Any fault files on lx?
yes
n 148
No
Locate the circuits on lx that have the lowest RP's which are less than c's and the same port types as was used on the normal route. (use the Circuit matching routine for this task.)
q 147
Open a TREE entry for the other end station of this link x is the parent station. Note the group that was used for the pre-emptable circuits used. Entries will only be made at this station for circuits that have a pre-emptable circuit to this station.
If lx is a satellite link, have any pre-emptable circuits been found over it?
no
r 150
Yes
s 149


Read & link to all FF's for lx.
n 148
Is there a link level failure on lx?
Yes
o 149
No
Are the channel or group level failures on lx?
No
q 147
Yes
Delete the CCSD's on lx that are on failed groups or channels in the core copy of lx's LF.
Any CCSD's in the core copy of lx's LF?
No
o 149
Yes
q 147


s
149
o
149
Any status IP stations in TREE?
Yes
No
Delete lx from the link catalog at station x and dump its LF.
Select an IP entry and calculate COST for the segment from x to this station for the highest RP circuit. Add the cost to x to get total known altroute segment cost.
Is lx a satellite link ?
yes
E
150
no
t
146
For the same entry, calculate HITS & HTS and enter into TREE.
Does any IP entry have the same station as another CLOSED, IP or OPEN entry?
yes
Delete the member of the pair with highest (2*COST+HITS+HTS) for the highest RP circuit entry at that station.
no
Label search status as OPEN for all circuits in the entry having IP status.
Label the remaining member as OPEN for all circuits in the entry.


Read the SF for the station sx on the other end of the satellite link.
r 150
Create a link catalog of satellite links from sx. Do not list totally failed links or the link from x.
Dump the SF of sx.
E 150
Is the link catalog at sx empty?
s 149
yes
No
Read the LF for a link (lx) in the catalog.
v 152


X
151
Read & link to all FF's for lx.
Is there a link level failure on lx?
Delete lx from the link catalog and dump its LF.
E
150
Yes
No
Are the channel or group level failures on lx?
D
152
No
Yes
Delete the CCSD's on lx that are on failed groups or channels in the core copy of lx's LF.
Any CCSD's in the core copy of lx's LF?
Delete lx from the link catalog and dump its LF.
E
150
No
Yes
D
152



v 152
no
Any fault files on lx?
yes
x 151
No
Locate the circuits on tx that have the lowest RP's which are less than c's and the same port types as was used on the normal route. (Use the circuit matching routine for this task.)
D 152
Open a TREE entry for the other end station of this link x is the parent station. Note the group that was used for the pre-emptable circuits used. Entries will only be made at this station for circuits that have a pre-emptable circuit to this station. Enter lx as the satellite link from which the circuits found are to be moved. Note the satellite group and channel numbers of the moved circuits.
s 149

INTENTIONALLY LEFT BLANK

INTENTIONALLY LEFT BLANK

INTENTIONALLY LEFT BLANK

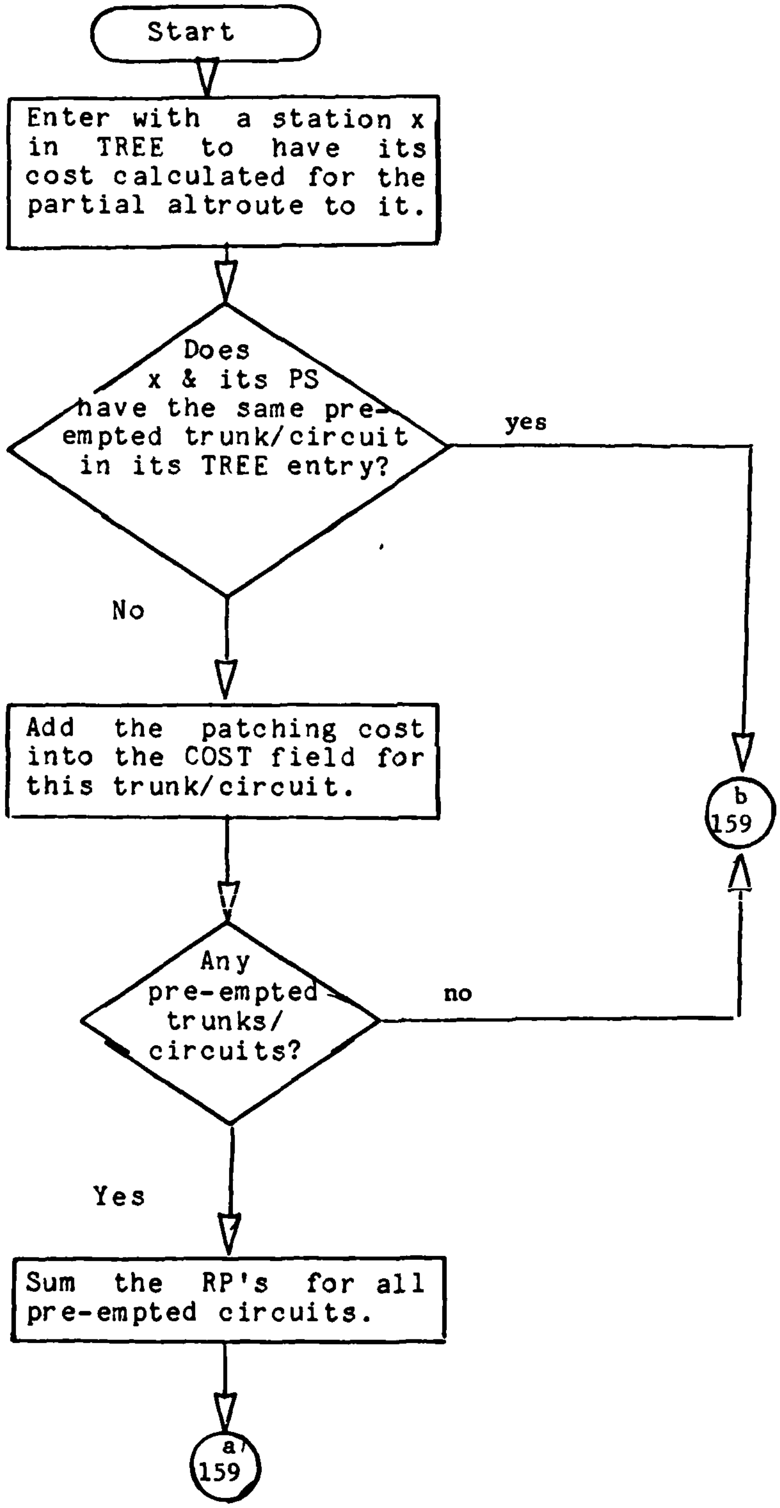
INTENTIONALLY LEFT BLANK

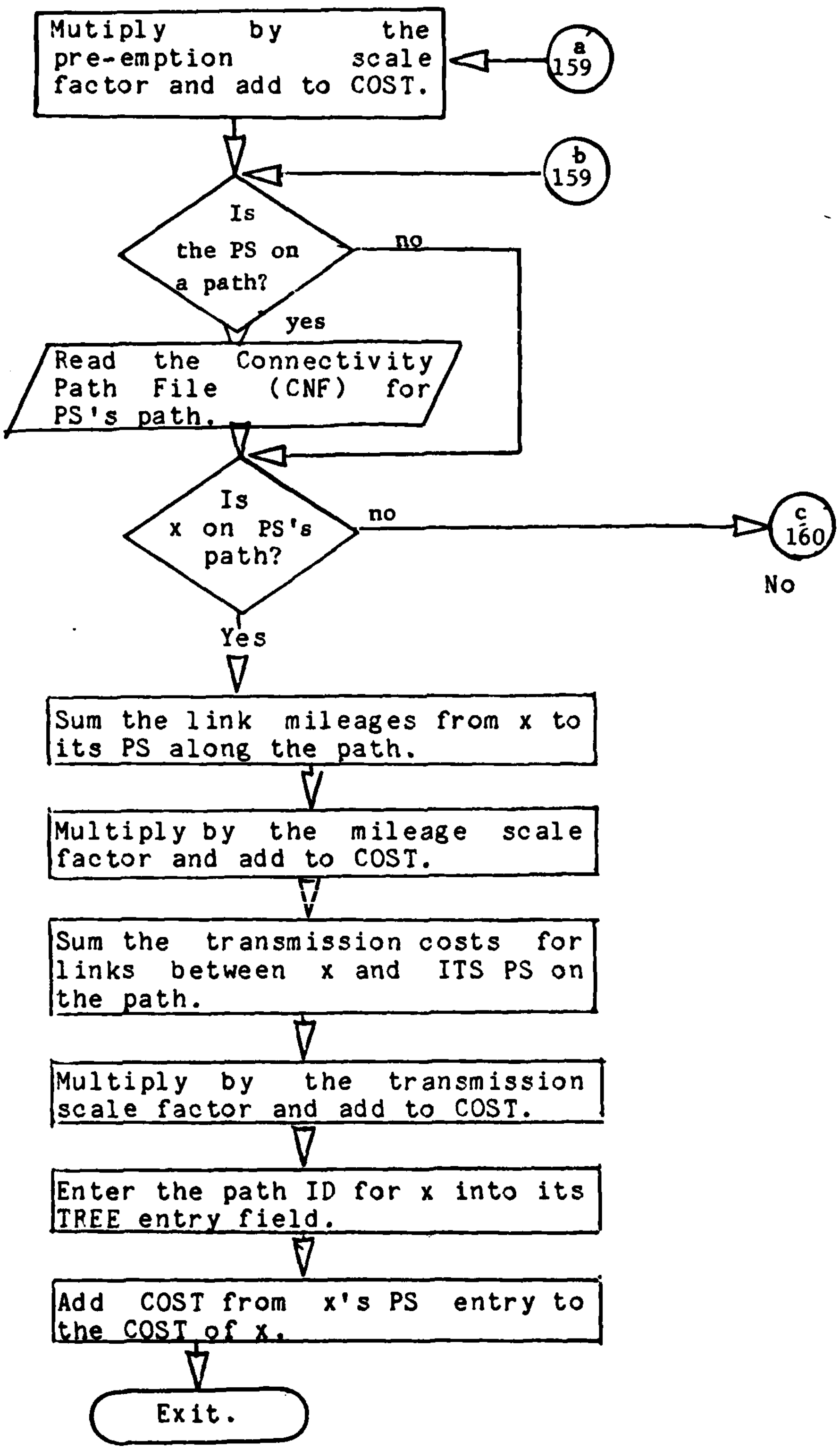
2.7.5 The Cost Calculation Routine

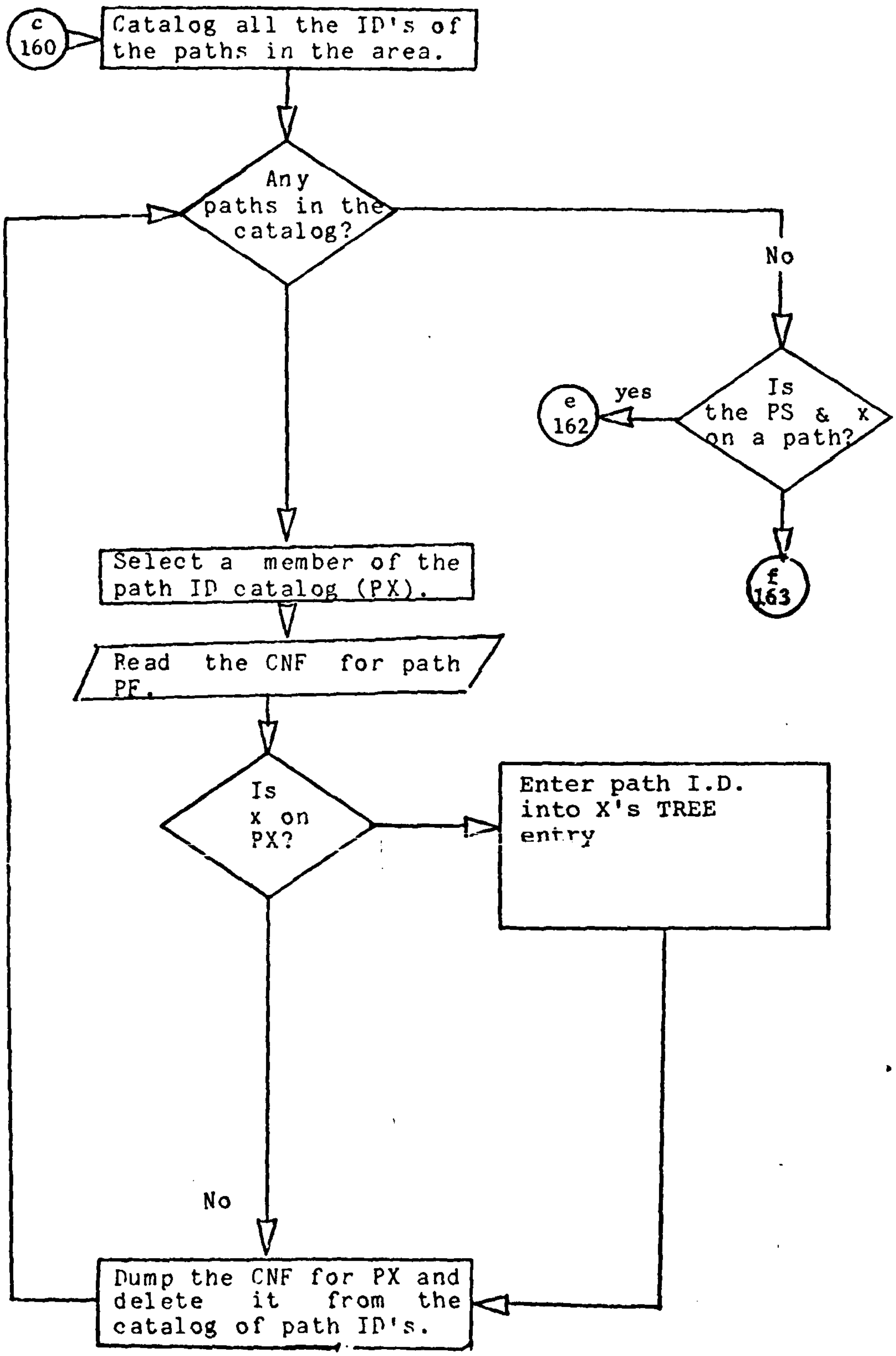
2.7.5.1 Discussion of the Modifications--The cost calculating routine sees one major change in a CRU network. The paths that the parent station and current search station will always be connected paths if both stations are on identified paths. This is true because the two stations are only separated by one link and one link cannot hop over an entire intermediate path.

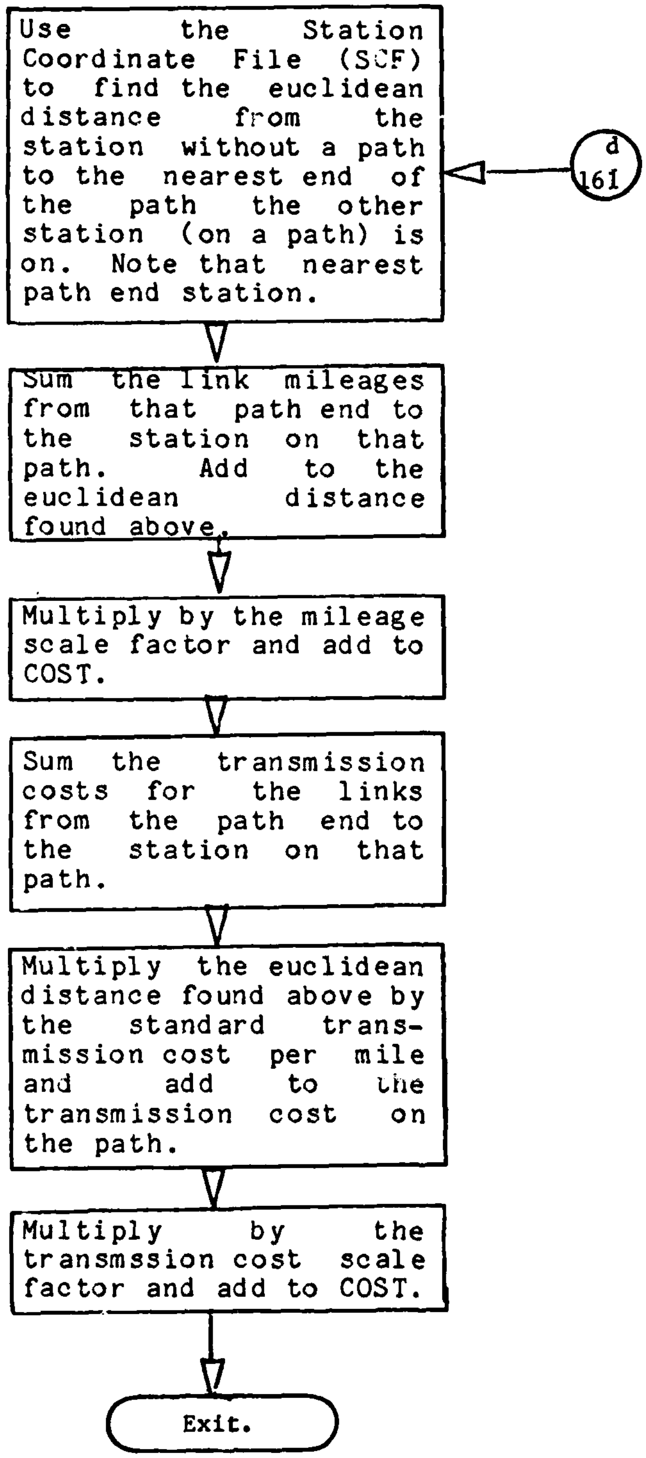
2.7.5.2.

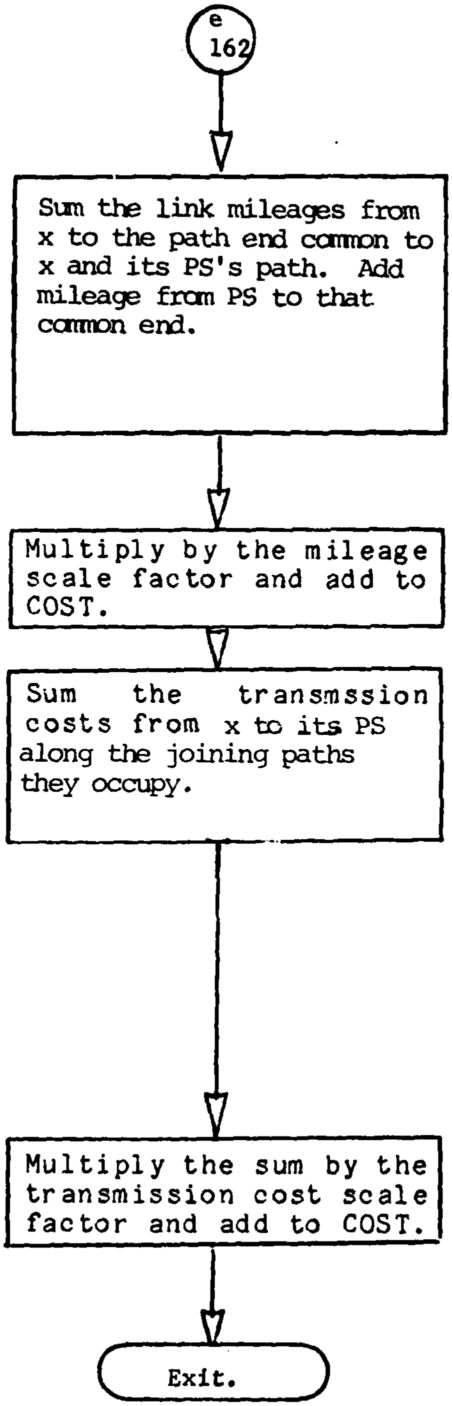
Modifications to the Cost Calculating Routine

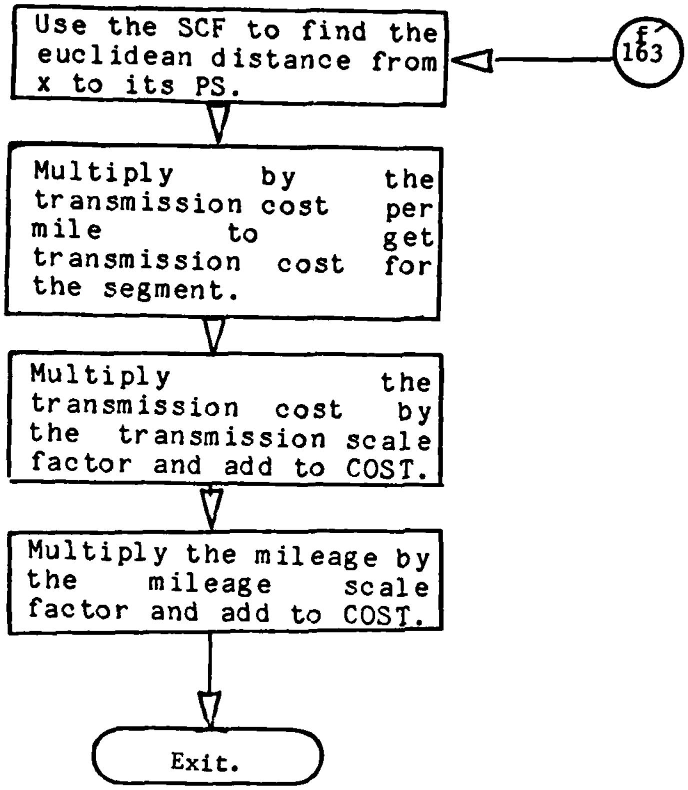



Mutiply by the pre-emption scale factor and add to COST.
a 159
b 159
Is the PS on a path?
no
yes
Read the Connectivity Path File (CNF) for PS's path.
Is x on PS's path?
no
c 160
No
Yes
Sum the link mileages from x to its PS along the path.
Multiply by the mileage scale factor and add to COST.
Sum the transmission costs for links between x and ITS PS on the path.
Multiply by the transmission scale factor and add to COST.
Enter the path ID for x into its TREE entry field.
Add COST from x's PS entry to the COST of x.
Exit.

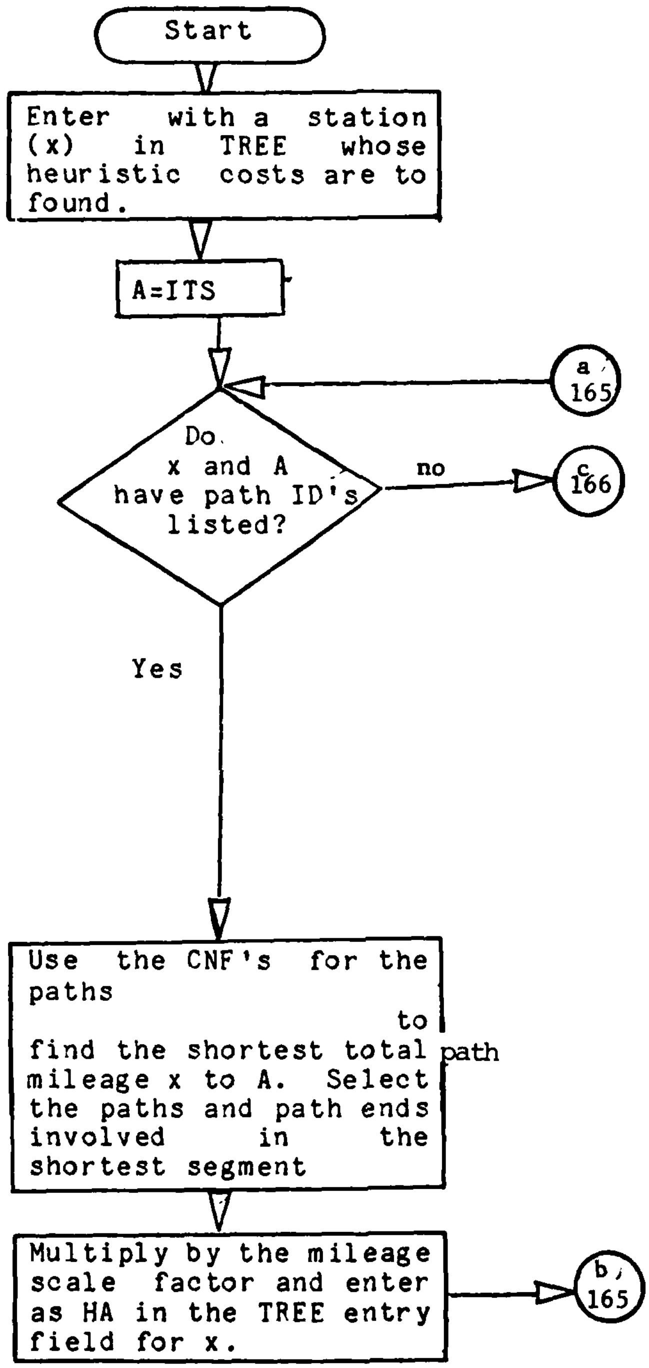

c 160
Catalog all the ID's of the paths in the area.
Any paths in the catalog?
No
Is the PS & x on a path?
yes
e 162
f 163
Select a member of the path ID catalog (PX).
Read the CNF for path PF.
Is x on PX?
Enter path I.D. into X's TREE entry
No
Dump the CNF for PX and delete it from the catalog of path ID's.

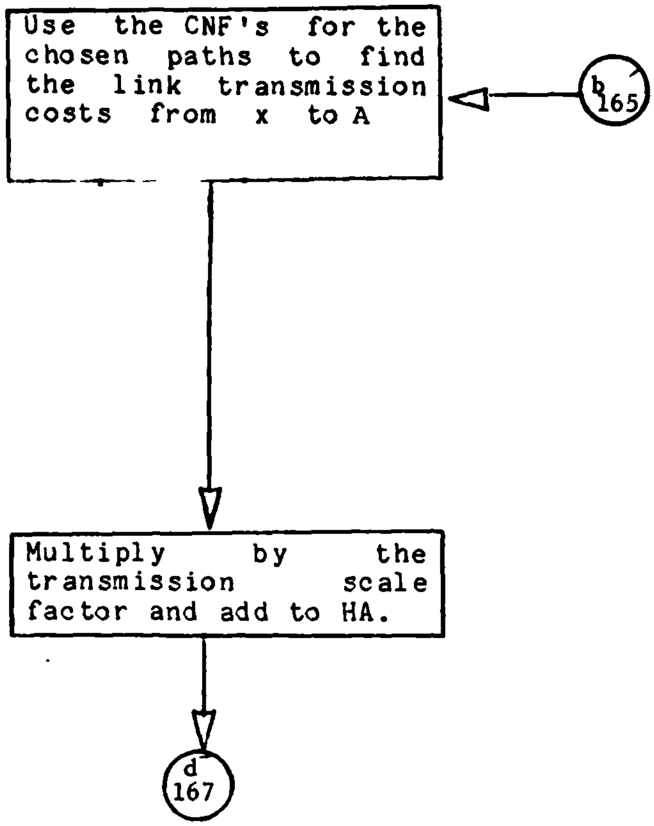

Use the Station Coordinate File (SCF) to find the euclidean distance from the station without a path to the nearest end of the path the other station (on a path) is on. Note that nearest path end station.
d 161
Sum the link mileages from that path end to the station on that path. Add to the euclidean distance found above.
Multiply by the mileage scale factor and add to COST.
Sum the transmission costs for the links from the path end to the station on that path.
Multiply the euclidean distance found above by the standard trans- mission cost per mile and add to the transmission cost on the path.
Multiply by the transmssion cost scale factor and add to COST.
Exit.

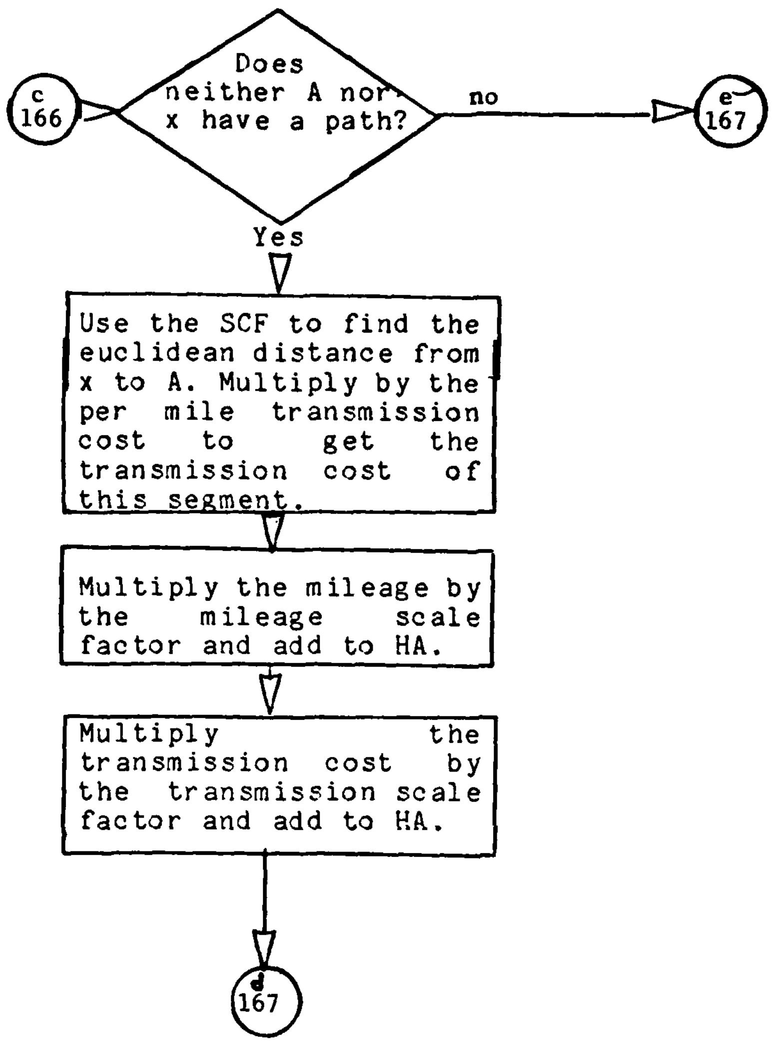

e
162
Sum the link mileages from x to the path end common to x and its PS's path. Add mileage from PS to that common end.
Multiply by the mileage scale factor and add to COST.
Sum the transmssion costs from x to its PS along the joining paths they occupy.
Multiply the sum by the transmission cost scale factor and add to COST.
Exit.

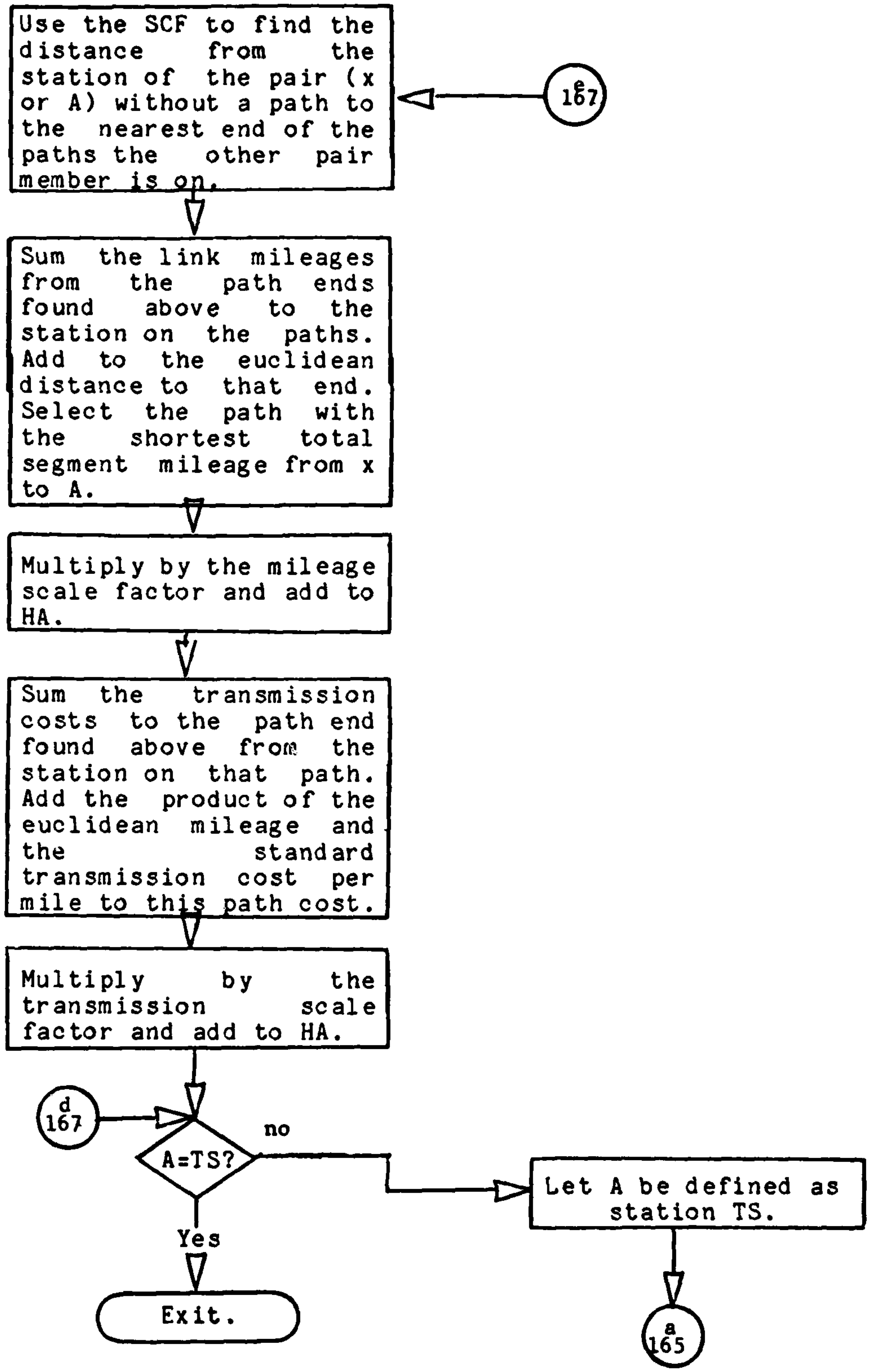

Use the SCF to find the euclidean distance from x to its PS.
f
163
Multiply by the transmission cost per mile to get transmission cost for the segment.
Multiply the transmission cost by the transmission scale factor and add to COST.
Multiply the mileage by the mileage scale factor and add to COST.
Exit.

2.7.5.3 <u>The Heuristic Cost Calculation Routine</u>--




Use the CNF's for the chosen paths to find the link transmission costs from x to A
b
165
Multiply by the transmission scale factor and add to HA.
d
167


c
166
Does
neither A nor
x have a path?
no
e
167
Yes
Use the SCF to find the euclidean distance from x to A. Multiply by the per mile transmission cost to get the transmission cost of this segment.
Multiply the mileage by the mileage scale factor and add to HA.
Multiply the transmission cost by the transmission scale factor and add to HA.
d
167

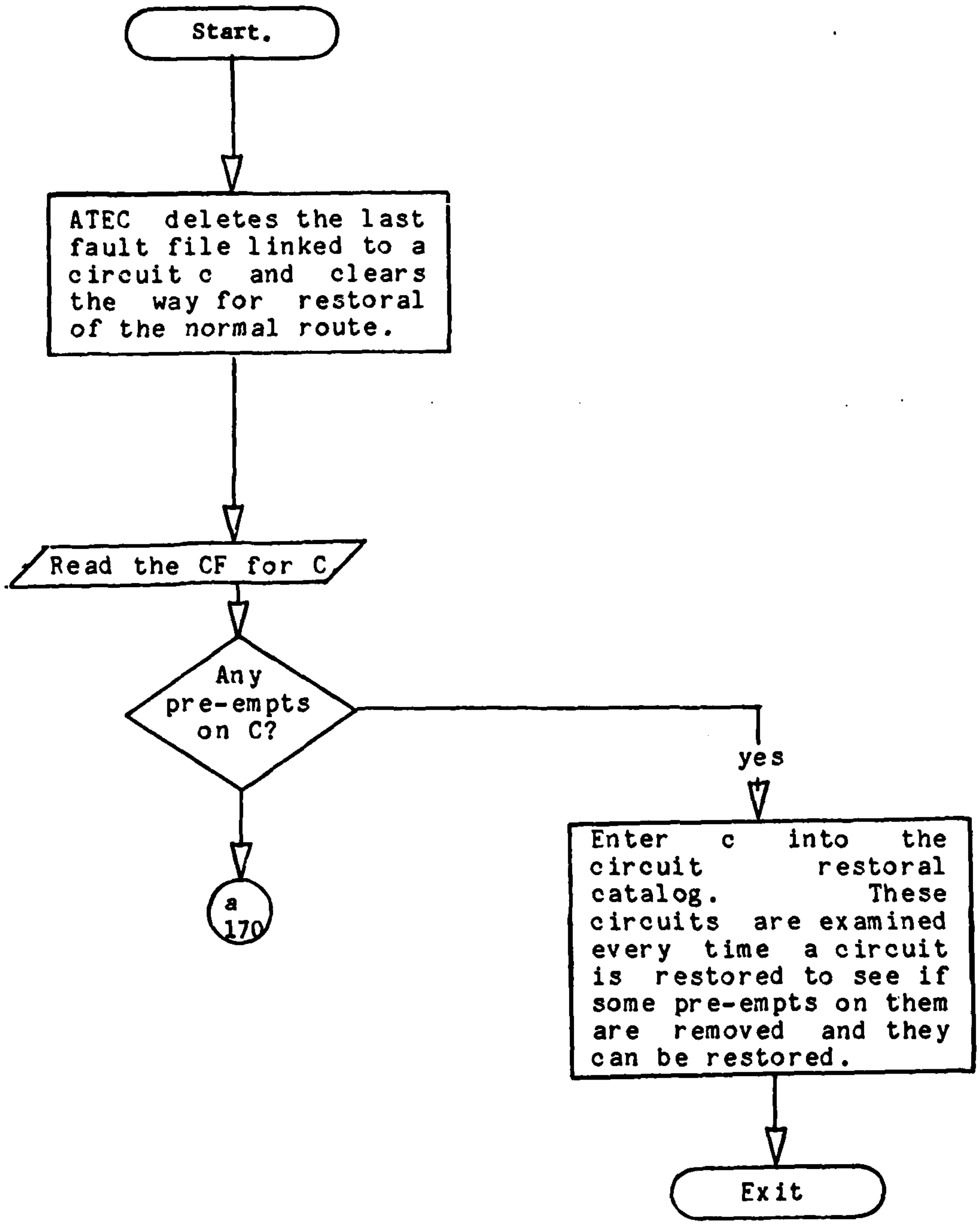

Use the SCF to find the distance from the station of the pair (x or A) without a path to the nearest end of the paths the other pair member is on.
e 167
Sum the link mileages from the path ends found above to the station on the paths. Add to the euclidean distance to that end. Select the path with the shortest total segment mileage from x to A.
Multiply by the mileage scale factor and add to HA.
Sum the transmission costs to the path end found above from the station on that path. Add the product of the euclidean mileage and the standard transmission cost per mile to this path cost.
Multiply by the transmission scale factor and add to HA.
d 167
A=TS?
no
Let A be defined as station TS.
Yes
Exit.
a 165

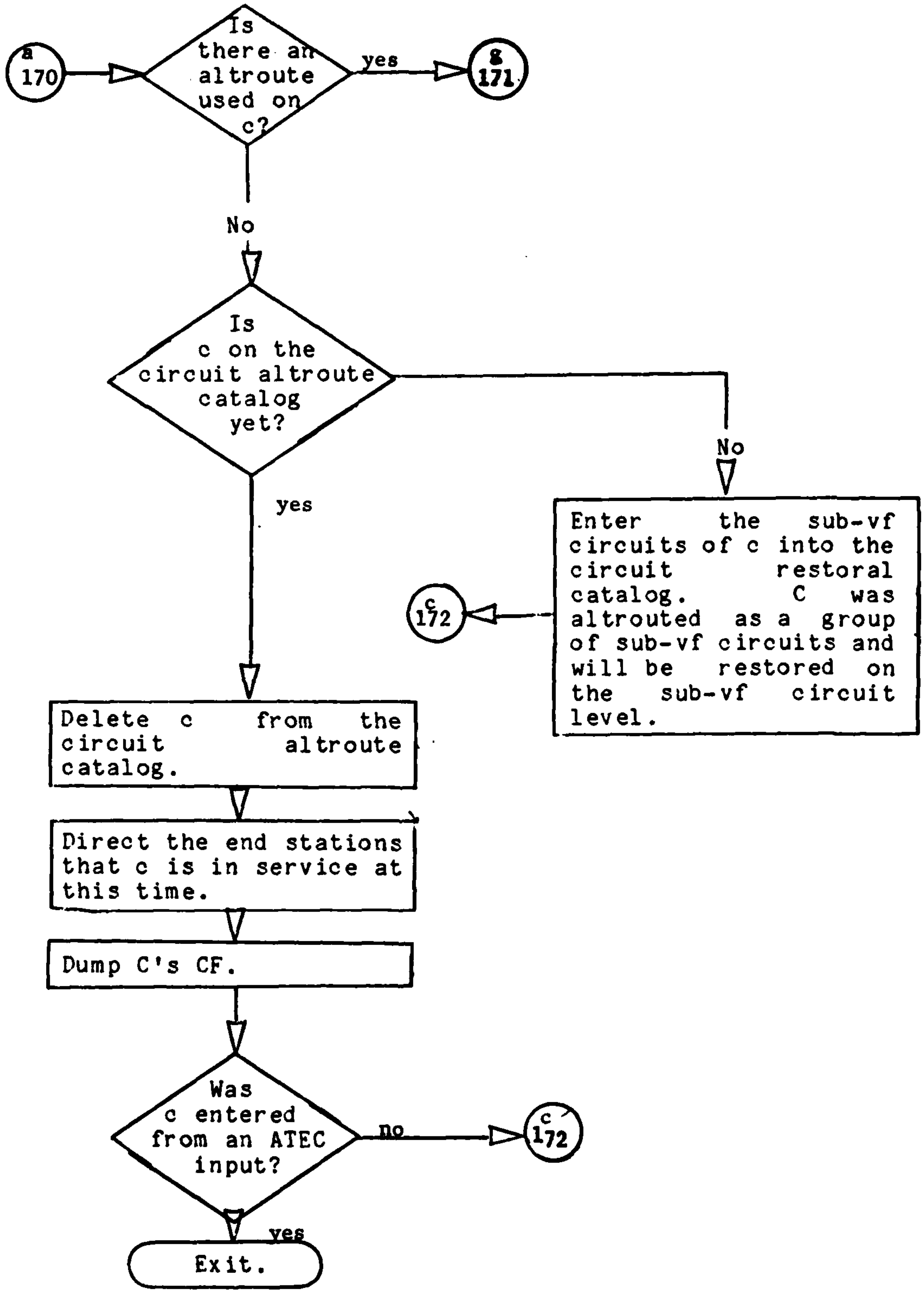
2.7.6 The Restoral Routine

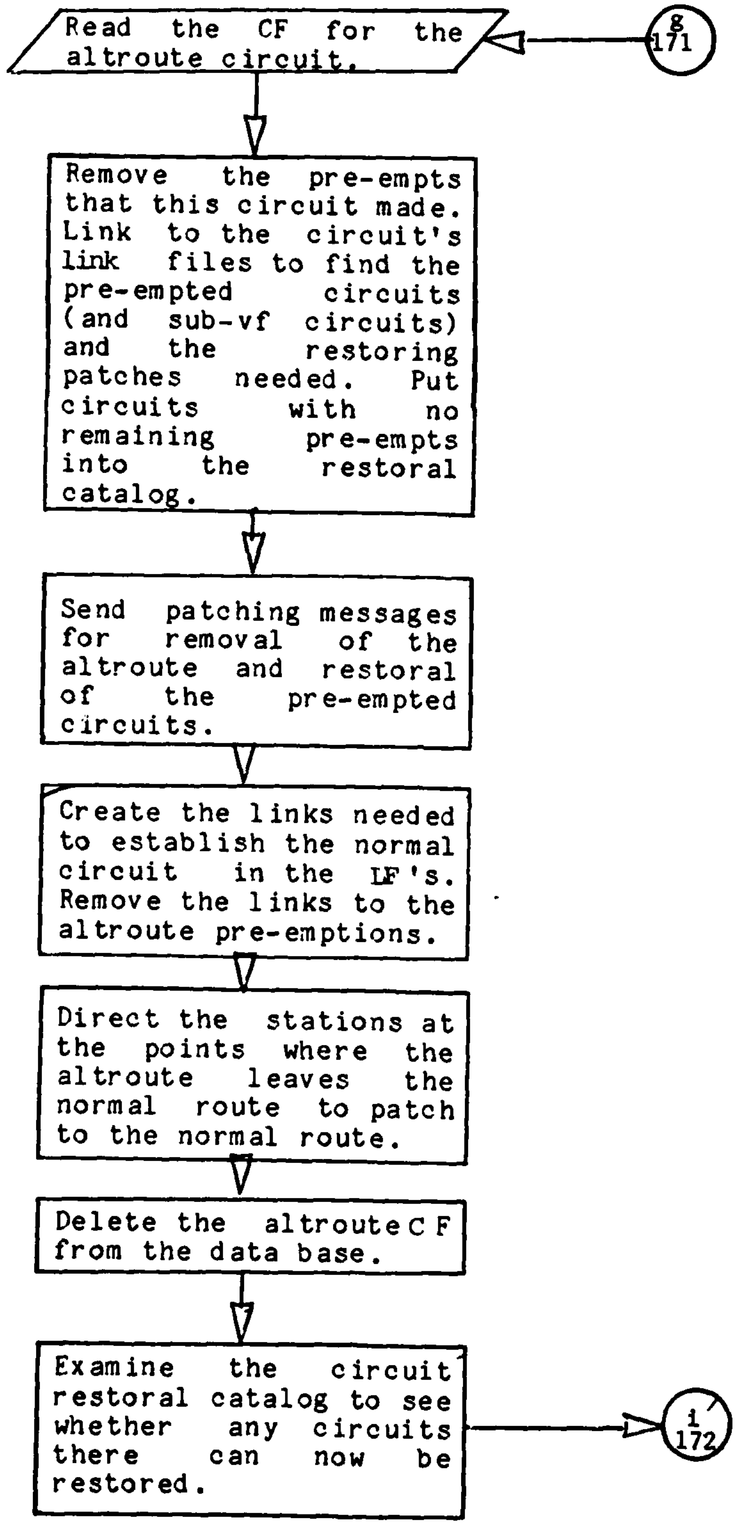
2.7.6.1 Discussion of the Modifications--The obvious change to this routine is the elimination of the restoral of trunks - they no longer exist in the service hierarchy.

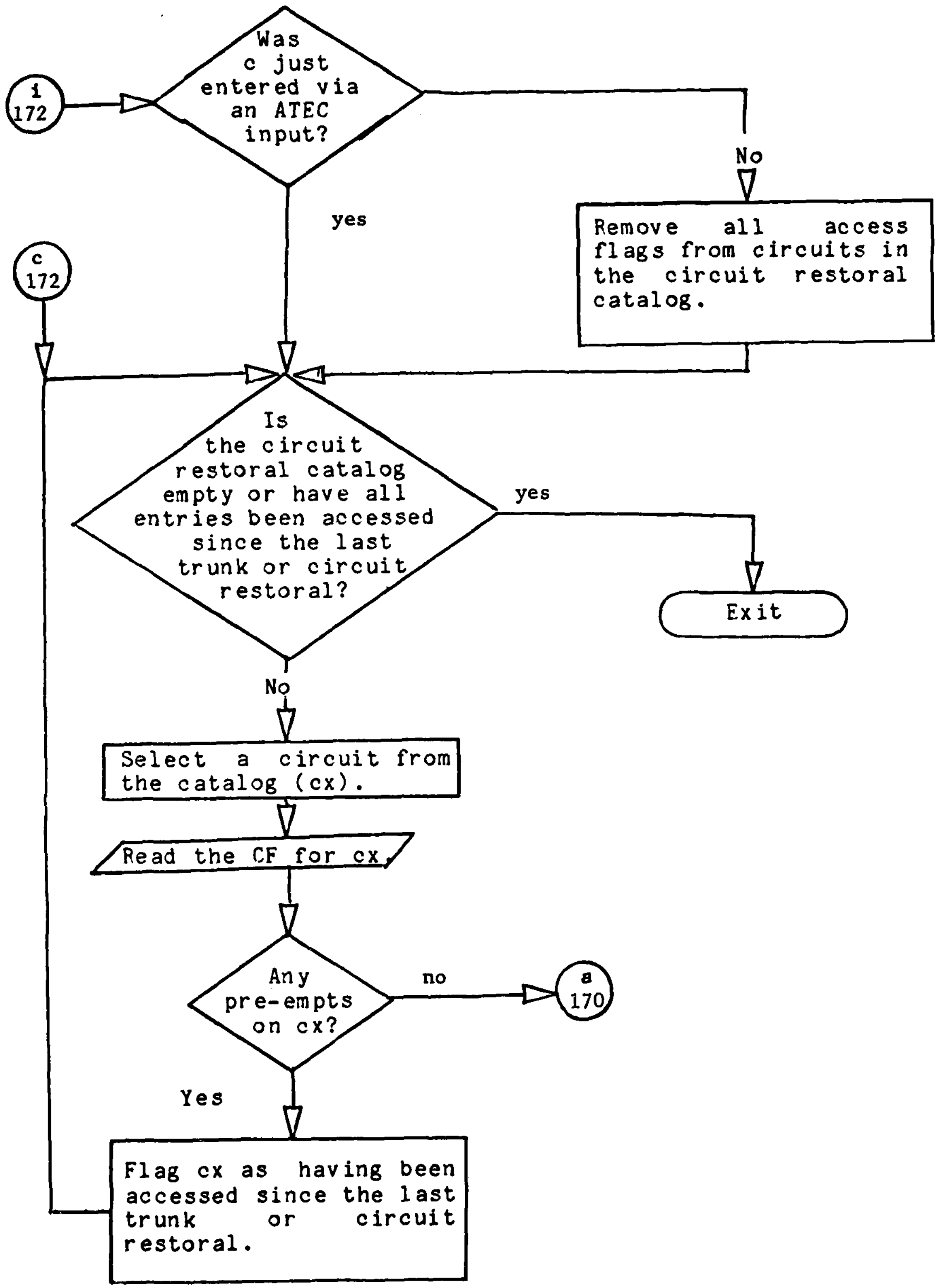
There is no longer the double linking of pre-emptions from trunk to circuits - circuits are the only level of service that can pre-empt other circuits. Sub-vf circuits still are present in the network and will be double linked to pre-empted vf circuits they are carried on.

2.7.6.2.
Modifications to the Restored Routine.




a
170
Is there an altroute used on c?
yes
g
171
No
Is c on the circuit altroute catalog yet?
No
Enter the sub-vf circuits of c into the circuit restoral catalog. C was altrouted as a group of sub-vf circuits and will be restored on the sub-vf circuit level.
c
172
yes
Delete c from the circuit altroute catalog.
Direct the end stations that c is in service at this time.
Dump C's CF.
Was c entered from an ATEC input?
no
c
172
yes
Exit.


Read the CF for the altroute circuit.
g
171
Remove the pre-empts that this circuit made. Link to the circuit's link files to find the pre-empted circuits (and sub-vf circuits) and the restoring patches needed. Put circuits with no remaining pre-empts into the restoral catalog.
Send patching messages for removal of the altroute and restoral of the pre-empted circuits.
Create the links needed to establish the normal circuit in the LF's. Remove the links to the altroute pre-emptions.
Direct the stations at the points where the altroute leaves the normal route to patch to the normal route.
Delete the altroute CF from the data base.
Examine the circuit restoral catalog to see whether any circuits there can now be restored.
i
172


i 172
Was c just entered via an ATEC input?
No
yes
Remove all access flags from circuits in the circuit restoral catalog.
c 172
Is the circuit restoral catalog empty or have all entries been accessed since the last trunk or circuit restoral?
yes
Exit
No
Select a circuit from the catalog (cx).
Read the CF for cx.
Any pre-empts on cx?
no
a 170
Yes
Flag cx as having been accessed since the last trunk or circuit restoral.

2.8 RESPONSE TIME ANALYSIS

The objective of the response time analysis is to provide inputs to the benefits analysis. The response time analysis was performed to compare the relative time associated with performing fault/stress detection, fault isolation, and restoral action execution on the DCS in the 1985 time frame for different degrees of automation. The emphasis of this analysis has been directed toward the analysis of transmission system control algorithms. The specific control analysed is associated with connectivity restoral and reconfiguration in response to stresses in the transmission system. The analysis includes an investigation of present manual procedures, generation of restoral plans, reconfiguration actions, fault detection and isolation, the role of system control (i.e., ATEC) and information flow to various levels in the hierarchy.

The altroute searching algorithm previously described together with the deployment of the Channel Reconfiguration Unit (CRU) are the tools that allow automation. The purpose of the algorithm is to generate, in real time, altroute plans to reestablish network connectivity. The capability this algorithm provides can be thought of as replacing or augmenting the requirement for ATEC to store restoration plans at both the Sector and Node. The algorithm can reside at either area or sector. The algorithm will find application in generating the plans that are stored at the node or sector as well as responding to requests for special stress related rerouting and restoration plans. The CRU on the other hand will eliminate manual patching when reconfiguration of network assets is required for connectivity restoration.

In normal operation "reroute or altroute" plans will be generated and sent to ATEC as required (as dictated by policy and procedures). It is assumed that the steady state, normal day-to-day requirements for this altrouting algorithm are related to system connectivity and TSO implementation and the associated requirements for the concomitant restoration plan storage. In other words, as a new circuit is implemented that requires a reroute plan (because of restoration priority for instance) then the area system control processor can or will provide the plan to ATEC. It is further assumed that ATEC will keep track of the status and viability of each plan that it has stored.

Alternately, the requirement for storage of restoral and reroute plans can be eliminated and the algorithm can provide the needed information for dissemination as dictated by the conditions at hand. This second alternative appears to be the most desirable because the on going maintenance of stored pre-planned altroutes is a significant activity, whereas the real-time generation of altroute plans takes advantage of current conditions in the network and will take less time overall in regard to processor work load.

Three conditions were analyzed to obtain the relative response times associated with different degrees of automation.

- Faults manually or ATEC detected and isolated, confirmed manually with manual altroute search and manual restoral/patching.

- Faults manually or ATEC detected and isolated, confirmed manually with automated altroute search and plan dissemination and manual restoral/patching.

- Faults manually or ATEC detected and isolated, confirmed manually, with automated altroute search and automatic patching restoration by using a channel reconfiguration unit (CRU) controlled by higher level system control.

2.8.1 Analysis Approach and Assumptions

The approach used in the analysis included a review of the procedures, policies, directives, and analysis of current restoration practices. This review coupled with assumptions related to the baseline system defined previously as being deployed in 1985, the modes of failure, the role of ATEC and system control, and fault isolation form the basis for estimating manual restoration timing, and processor and telementry system delays. These considerations are discussed in more detail below.

2.8.1.1 Baseline system--The baseline system is the deployment model previously developed and updated during an earlier phase of the study. This deployment model represents the European DCS Backbone in 1985 and is shown in Figure 2-11. This model has the Digital European Backbone fully deployed and operational. A functional block diagram of the digital transmission system is shown in Figure 2-12. The ATEC system is assumed to be deployed according to the hierarchy shown in Figure 2-13. This baseline was chosen rather than an abstract modeling approach because an abstract model could not highlight the system control problems caused by the peculiarities of the DCS. As indicated in previous reports, the European backbone was chosen because it is at least as complex as any other segment of the DCS and contains examples of every type of subsystem used in the DCS. The European area is reflective of user and mission objectives world wide and system control algorithms analyzed as applied to the European area can be directly extended and applied to other segments of the DCS.

AD-A080 767 HONEYWELL SYSTEMS AND RESEARCH CENTER MINNEAPOLIS MN F/G 17/2
SYSTEM CONTROL FOR THE TRANSITIONAL DCS.(U)
MAY 79 F C ANNAND, D E DOTY, R V KRZYZANOWSKI DCA100-78-C-0017
UNCLASSIFIED TR-3 SBIE-AD-E100 328 NL

3 OF 4
AD
A080 767

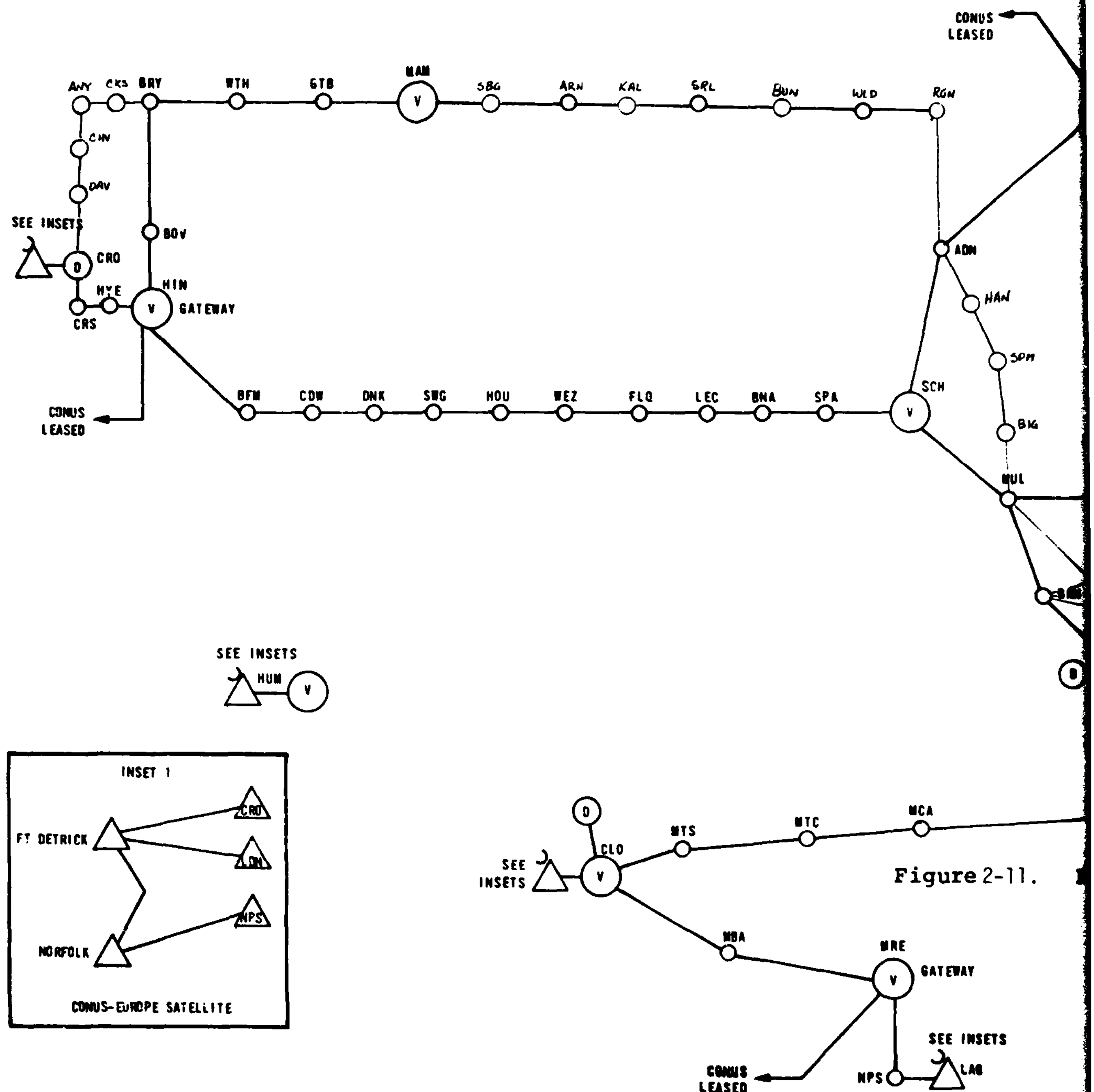
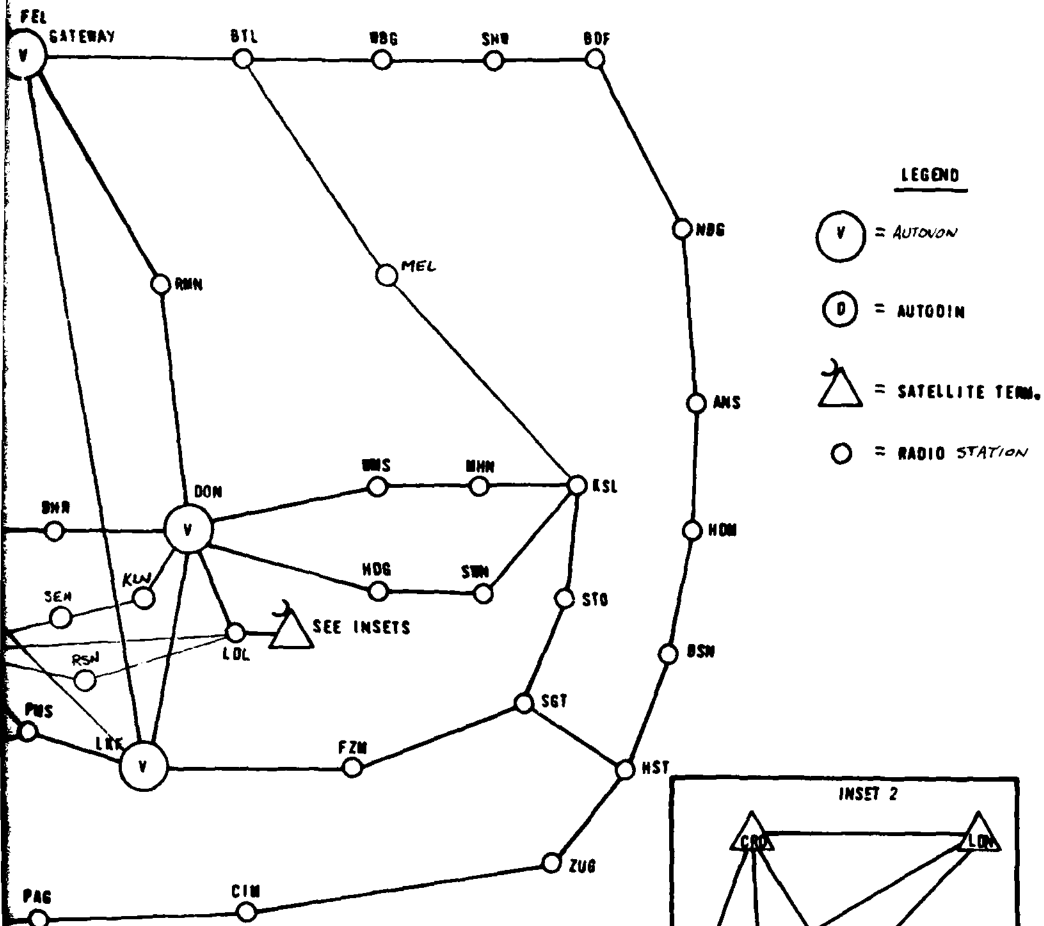
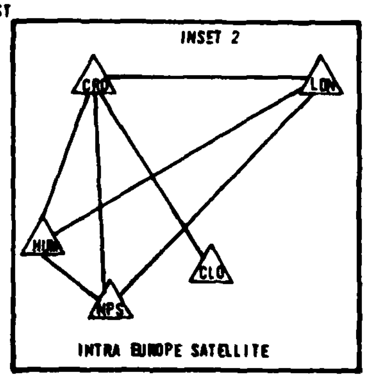

CONUS LEASED
ANY
CKS
BRY
WTH
GTB
MAM
V
SBG
ARN
KAL
SRL
BUN
WLD
RGN
CHN
DAV
SEE INSETS
D
CRO
BOV
HYE
HIN
GATEWAY
CRS
ADN
HAN
SPM
SCH
BIG
CONUS LEASED
BFM
CDW
DNK
SWG
HOU
WEZ
FLQ
LEC
BNA
SPA
MUL
SEE INSETS
HUM
INSET 1
FT DETRICK
CRO
LDN
NORFOLK
NPS
CONUS-EUROPE SATELLITE
D
CLO
MTS
MTC
MCA
SEE INSETS
MBA
MRE
GATEWAY
SEE INSETS
CONUS LEASED
NPS
LAB

Figure 2-11.





ropean DCS Backbone Model 1982-85

DIGITAL TRANSMISSION SYSTEM FUNCTIONAL BLOCK DIAGRAM

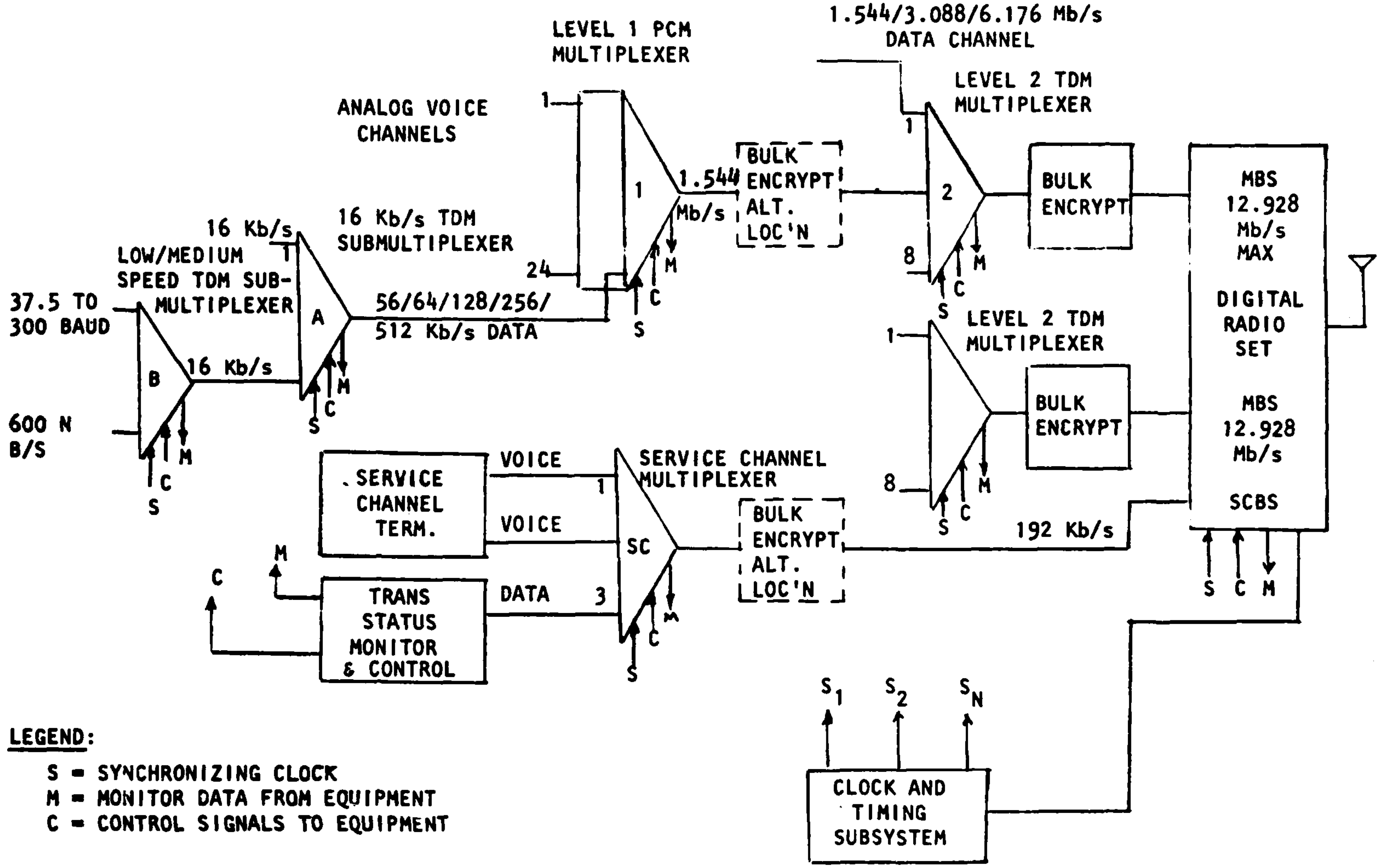


Figure 2-12. Digital Transmission System Functional Block Diagram

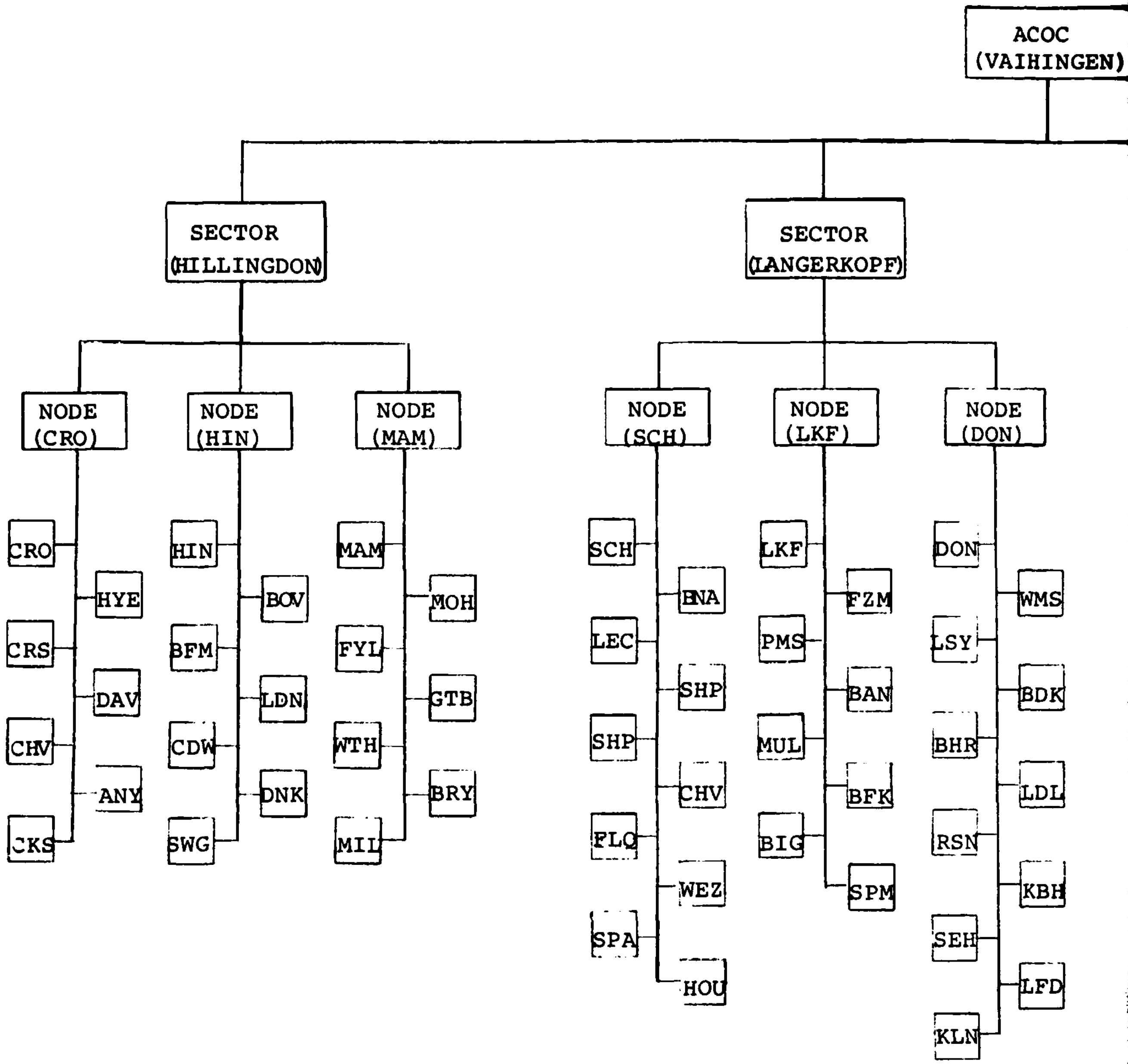
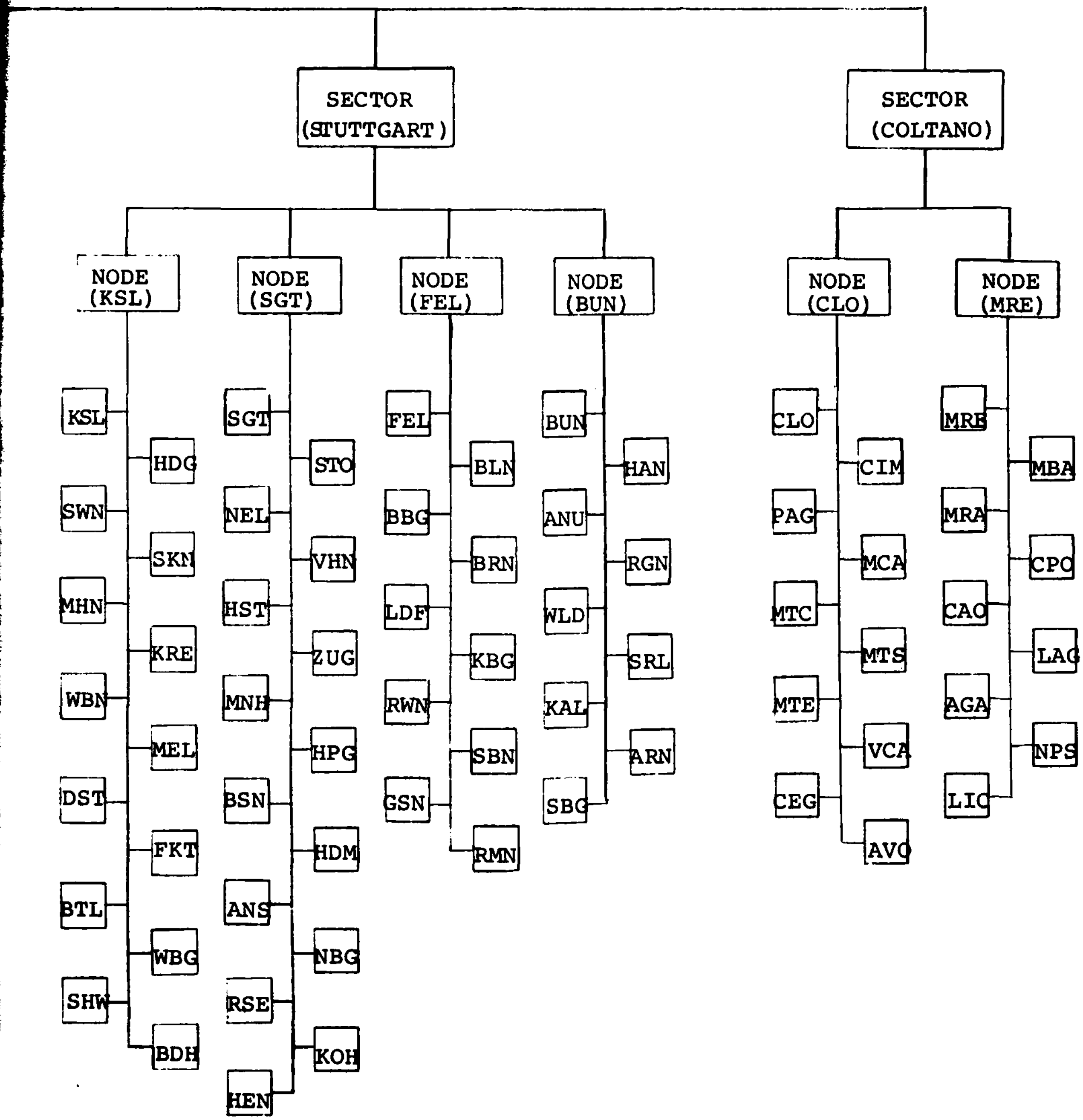


Figure 2-13. European ATEC



Deployment Hierarchy (1982-1985)

2.8.1.2 Scenario Selection--The digital radio sets and level 2 TDM multiplexers are virtually fully redundant and the bulk encryption unit has a clear mode bypass, therefore single failures in these subsystems will not create a stress requiring network reconfiguration or restoral by altroutes using the altroute searching algorithm. The failure of a level 1 PCM multiplexer will require circuit altroutes for the high priority users. A failure which will cause a fairly complex restoral action and reconfiguration is desired to determine the response time of the control algorithm and automatic patching/switching. Single circuit failures or even a level 1 mux failure will require alroute but cause no major upheaval in the network as they will likely be restored on the same link they were originally routed over. The type of failure selected for the analysis is a link failure wherein different routes must be found for restoration. The cause of the stress is not important; whether it be an act of God such as a wind storm, covert action such as jamming or operator error , it is the abnormal, pathological stress such as these that will require major network reconfiguration.

2.8.1.3 Single Circuit Failure Analysis--The response time for failure restoration has been investigated to the extent possible. Several references (e.g. references 11 and 12) relating to technical control functions have provided the timing associated with tech controller actions and from these can be derived the timing of restoration and altrouting single circuit failures. The response times derived provide the basis for extrapolation to larger stresses with the appropriate assumptions. The conclusions drawn in the references are based on actual observations over extended periods of time at a number of TCF's in Europe. The conclusions that are germane to this analysis are:

- The mean time spent by a tech controller associated with each circuit outage was 10 minutes. This time included the fault isolation, coordination,testing, patching, rerouting, and reporting action.

- The average elapsed time from

  - user call to TCF first action = 1-2 min

  - from first action to decision to altroute = 6.8 minutes

The profile of a single circuit failure relative to technical control action is shown in Table 2-3.

The time required to restore a circuit outage by altroute action was not specifically stated. Of all the circuit outages analyzed (about 300),

only about 10% were restored by altrouting. This means that outages requiring reroutes will take considerably longer than 10 minutes while outages where the problem is cleared, resolved or handed off to another station or maintenance took slightly less than 10 minutes. The time to a decision to altroute was found to be 8.0 minutes but of the remaining actions (altroute search, coordination, patching, testing, and reporting) only the patching time can be derived from previous analysis. The patching time for 31 reroutes was found to be an average of 1.2 minutes. The remaining actions were analyzed and reasonable times assigned based upon actual experiences in the field. The results of this response time synthesis is shown in table 2-4. The table shows that the circuit is actually restored in 12.2 minutes while the user would count the outage time as 13.2 minutes.

In equation form, the response time from outage to reporting restoration complete is as follows:

$$\begin{aligned} RT &= FI + AS + C + CP + PT + UC + RPT \\ &= 8.0 + 1.0 + 1.0 + 1.2 + 1.0 + 1.0 + 1.0 \\ &= 14.2 \text{ minutes} \end{aligned}$$

where RT = Restoration response time
FI = Fault isolation and decision to altroute time
AS = Altroute search time (manually)
C = Coordination of altroute planning time
CP = Circuit patch action time
PT = Patch test and checkout time
UC = User coordination time
RPT = Reporting time

2.8.1.4 Response Time Considerations for Larger Failures--As a first step in the extrapolation of the known response times for restoration of a single circuit to restoration caused by a major failure, a scenario involving the present DCS in Europe was analyzed. A single link failure between Feldberg and Langerkopf was chosen since we had a fairly recent copy of the DCA data base for that portion of the DCS. This scenario was developed using the present procedures for restoration adjusting the times as appropriate to account for multiple circuit reroutes, several technical controllers being involved at the affected stations, and other measures to expedite the restoration process such as concurrent restoration at several sites. The next step in the analysis consisted of extrapolating the manual procedures to 1985, using the DCS Deployment Model developed earlier in the study and projected DCS connectivity for the same time frame. This analysis produced three scenarios described in the following section. The first scenario produces the basis for response time improvement by increased automation. The primary areas of automation analyzed were the replacement of the manual efforts associated with altroute search and finally

TABLE 2-3. PROFILE OF A SINGLE CIRCUIT FAILURE

	Fault Isolation	Alt Route Search	Coord.	Testing	Patching	Reporting
1. User call or alarm received on circuit failure						
2. Fault Isolation	x		x	x		
3. Decision to altroute	x		x			
4. Check National Communication System (NCS) restoration priority and CCSD		x				
5. Check reroute plans for preplanned route		x	x	x		
6. Check status of planned or possible reroute			x	x		
7. Coordination with connected TCF			x			
8. Notify losing user if preemption is involved			x			
9. Accomplish required patching					x	
10. Test electrical connection and adjust as required				x		
11. Coordination with connected TCF to verify completion			x			
12. Call user to resume traffic			x			
13. Fill out trouble ticket						x
14. Log action taken						x
15. Generate report						x

Table 2-4. Single Circuit Failure Restoration

Time from fault detection (user call) thru decision to altroute	8.0 minutes
Time allotted to altroute search	1.0
Time allotted to coordination with connected TCF	1.0
Time associated with implementing patch	1.2
Time associated with patch test and verification	1.0
Time allotted to coordination with circuit user	1.0
Time allotted to reporting	1.0
Total time	14.2 minutes

the replacement of the manual patching by the assumption that the channel reconfiguration unit (CRU) is fully deployed in the DCS-Europe. The two subsequent scenarios reflect this analysis where the manual altroute search time is replaced by an automated generation and dissemination of the patching instructions in the second and in the third, manual patching is replaced by commands sent to the CRU to affect the switching and restoration automatically. Figure 2-14 shows the flow chart depicting the path followed by the three scenarios in evaluating the restoration response time.

2.8.2 Response Time Scenarios

Three response time scenarios are presented that analyze the steps and time involved in accomplishing circuit/digroup restoration due to a link failure between two major technical control facilities. The link between Langerkopf (LKF) and Feldberg (FEL) was chosen for this analysis. Two of the reasons for choosing this link are that (1) we have a detailed listing of the circuits for these and the surrounding stations valid in the mid to late 1970's (current) and therefore realistic decisions regarding rerouting and preemptions could be made and be reasonably extrapolated to the 1985 time frame and that (2) the primary restoration/reconfiguration route (thru Donnersberg (DON) remains the same. Figure 2-15 shows the connectivity in the vicinity of the LKF-FEL link failure.

Using the above link data, the following assumptions are made for the current link between LKF and FEL. The link carries two supergroups (10 groups or 120 channels). Eight groups are made up in LKF, seven of which terminate in FEL, one is a thru group in FEL terminating in England (MAM). The two remaining groups are thru groups in both LKF and FEL. In 1985, the link carries a single mission bit stream (MBS) employing an eight port 2nd level multiplexer capable of carrying 192 channels. The eight di-groups (digital groups) are routed as follows: two are spare, five are made up in LKF and are broken out at FEL and one is a thru group in both stations (Kaiserslautern to Frankfurt).

The current (mid 1970's) restoration priority and distribution of the circuits on the 7 groups between LKF-FEL is extrapolated to the 5 di-groups in the 1985 time frame.

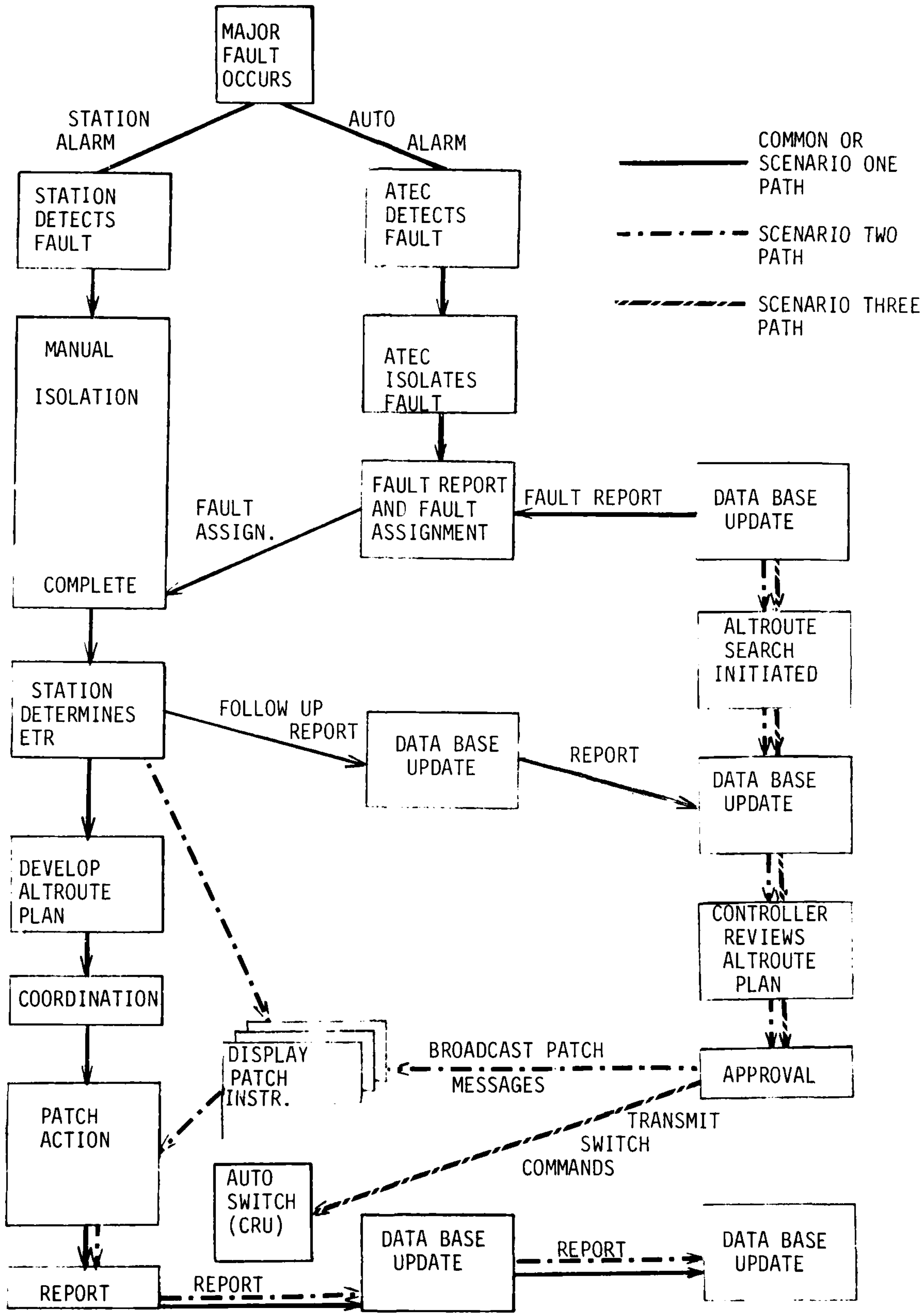


Figure 2-14. Restoration Response Time Flow Chart

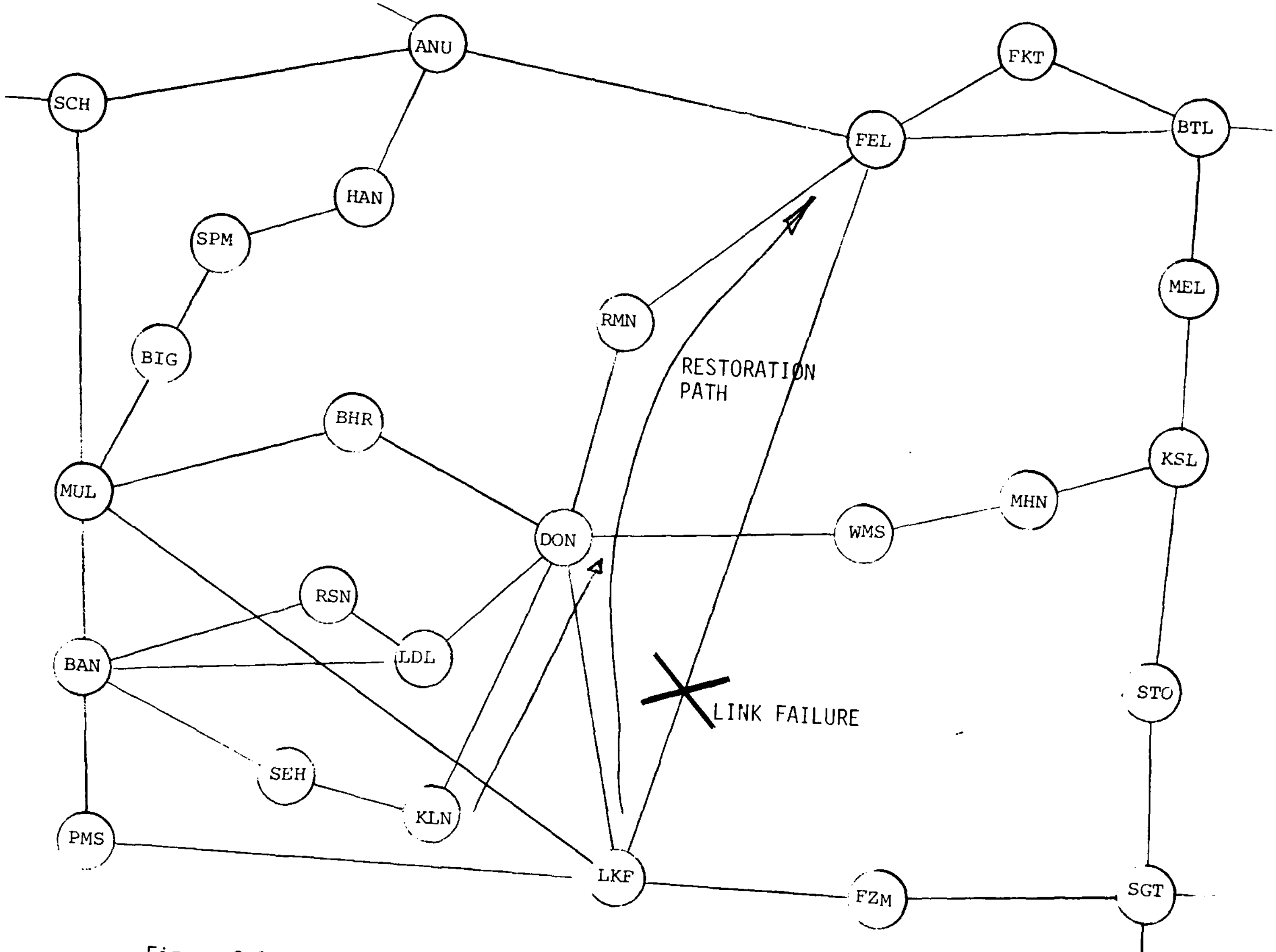


Figure 2-15. DCS Connectivity Near LKF - FEL Link Failure

Current distribution (7 groups = 84 circuits)

Restoration Priority	Number of Ckts	% of total
RP1	48	57
RP2	10	12
RP3	5	6
RPO	19	23
Spare	2	2
	84	

Extrapolated Distribution 1985 (5 digroups = 120 circuits)

Restoration Priority	Number of Ckts
RP1	68
RP2	14
RP3	7
RPO	28
Spare	3
	120

It was also observed in the current routing that two groups contained 10 RP1 circuits out of 12. This observation was extrapolated to 1985 resulting in the assumption that two of the digroups will contain as many as 20 RP1 circuits.

A similar analysis was performed on the LKF to DON link and the DON to FEL link via RheinMain (RMN), the primary altroute path. In the 1985 connectivity, there are no direct di-groups connecting LKF and FEL via DON but there are five digroups (including one spare) between LKF and DON and seven digroups (including two spares) between DON and FEL (all via RMN) that provide the means for restoration. Extrapolating the circuit distribution from the current time to the mid 1980's gives the following distributions:

1985 Distributions

	LKF-DON-(96 Ckts)		DON-FEL (120 Ckts)	
Restoration Priority	Current %	No. of Ckts in 1985	Current %	No. of Ckts in 1985
RP1	24	23	40	48
RP2	0	0	7	8
RP3	18	17	9	11
RPO	39	38	39	47
Spare	19	18	5	6
		96		120

It was further noted that two of the groups in the current routing between LKF-DON carried only one RP1 circuit and one of those also carried only two RP3 circuits. This was extrapolated to 1985 resulting in the assumption that one of the digroups between LKF and DON carries only two RP1 and four RP3 circuits.

2.8.2.1

Scenario One, Manual Altroute Search-Manual Restoration-- This scenario extends the present procedures for the restoration of circuits to the mid-1980's. The restoration philosophy is based primarily on the circuit restoration priority concept-namely an RP1A is restored before an RP1B or an RP2 is restored before an RP3 and so on down the list. Spares and the lowest priority circuits (RP00) are preempted first in the process. The process stops when the priority of the circuit to be restored equals the priority of the circuit to be preempted.

An additional capability exists in the all digital network which allows the patching of digroups to expedite restoration and reconfiguration caused by a major system stress.

A major stress has caused the link between LKF and FEL to fail. The cause of the failure is unimportant whether it be accidental, covert action or natural causes. The restoration scenario has been derived giving the sequence of events from which the restoration response time can be evaluated in terms of elapsed time. This scenario is shown in Table 2-5. For major failures such as this, ATEC does not aid in fault isolation since major station alarms have the affected station personnel alerted and working on the problem. At item number 5, the ETR is reported and the decision to altroute is made. From number 5 to number 9 a combination of coordination and altroute searching (planning) is accomplished, and 12 minutes elapse. The first restoration action consists of patching two digroups. These two patches restore 40 of the RP1 circuits. Between LKF and DON one spare digroup is used and one digroup is preempted. The preempted digroup adds two RP1 and four RP3 circuits to the restoration list. Between DON and FEL two spare digroups are used to complete the digroup patching and restoration. The timing associated with this action is derived from the single circuit failure/restoration analysis wherein 1.2 minutes for the patch and one minute for patch checkout is allotted for each digroup. The second digroup follows immediately between LKF and DON while the patching of the first di-group from DON to FEL is accomplished concurrently. The second digroup is patched from DON to FEL after it reaches DON and the total time elapsed to restore both digroups is 6.4 minutes. At item number 12, restoration of the RP1 circuits begins. Initially there were 68 RP1 circuits to restore; 40 were restored by the digroup patches and two were added to the list from the preempted digroup leaving 30 RP1 circuits. The timing to restore these circuits is derived

Table 2-5 SCENARIO ONE- SINGLE LINK FAILURE

Station Legend
A - LKF
B - FEL
C - DON
D - RMN
F - KLN

NO	TIME (min.)	STATION	
1	t = 0	A	Air Force Station A receives station alarms indicating loss of receive signal from Station B. ATEC also receives alarms; begins isolation/correction.
2	1	A	Radio maintenance advises Technical Control that the 192 channel receive from B is out.
3	3	-	ATEC assigns fault to LKF; issues fault report to AREA.
4	4	A	Radio maintenance detects problem to be a major failure and advises Technical Control that the estimated time to repair is 3 hours.
5	5	A	Station reports that estimated time to repair (ETR) is 3 hrs. Message forwarded by ATEC to AREA.
6	6	A	Technical Control contacts Station B and C to apprise them of the situation, that reroutes of critical circuits will be required, discuss approach in general terms.
7	15	A & B	Technical control reviews mux charts, circuit listings, in-house reroute plans and determines that this failure effects eight di-groups, 5 of which are made up in A and terminate in B and 1 is a thru group in both stations. Two di-groups are spare. The primary altroute path is then thru C & D, there are no direct digroup route from A to B thru C & D. There are 5 di-groups (including 1 spare) between A & C; there are 7 digroup (including 2 spares) between C & B.

Scenario One

Single Link Failure (continued)

NO	TIME	STATION	
8	17	A	Technical Control contacts B & C to outline approach to restore high priority circuits and critical users. Agreement is reached and restoration begins.
9	17	A	Technical Control concurrently contacts Station F, apprises him of failure and requests his assistance in accomplishing reroute of group that originates at his station due to the number of high priority circuits Station A must reroute.
10	21.4	A & C	Technical Control patches two di-groups, one on space and one preempts a digroup carrying 24 ckts all with priorities less than RP2 except for two RP1 circuits which will be restored individually. There is two digroup patches restore 40 RP1 circuits (primarily VON IST's).
11	23.4	C & B	Technical Control completes patching the two digroup between C & B using the two spare di-gr ups.
12	56.4	A,B&C	30RP1 ckts remain to be altrouted and are coordinated and patched individually according to priority. Spare and RPO ckts are used for this restoration.
13	83.9	A,B,&C	14 RP2 ckts are rerouted next. RPO ckts are preempted to reroute the RP2 ckts. 11 RP3 ckts are rerouted next. RPO ckts are preempted to reroute the RP3 ckts.
14	85.9	A	The reporting function continues concurrently with the restoration. The last circuit restored is reported at the time indicated.

from the single circuit restoration timing but modified to account for additional tech controllers vectored to support the restoration task. At LKF for instance, it is assumed the one technical controller is handling the coordination and work direction, a second tech controller is assigned to the reporting function, and two tech controllers are patching. Similar manning is applied to the task at DON and FEL. At DON, it is assumed that four tech controllers are assigned to patching, two working with LKF and two working with FEL. The relationship used to derive the timing for this item then is as follows:

$$TRC = \frac{NC}{NTC} (CP + PT)$$

Where
- TRC = time to restore circuits
- NC = number of circuits to be restored
- NTC = number of tech controllers assigned to patching
- CP = Circuit patch action time from single circuit analysis
- PT = Patch test/Checkout time

$$\text{therefore } TRC = \frac{30}{2} (1.2 + 1.0) = 33 \text{ minutes}$$

Similar reasoning applies to the 25 additional RP2 and RP3 circuits remaining at item 13 where 27.5 minutes are used.

2.8.2.2 Scenario Two, Automated Altroute Search Manual Restoration-- This scenario addresses the same circumstance as scenario one with the manual altroute search performed by the tech controllers replaced by altroute information generated and disseminated by the system control elements (see Table 2-6). At item number 3 when ATEC has isolated and assigned the fault to LKF, the automated altroute search algorithm is initiated. At number 6, the altroute plan is complete and can be reviewed by the system control controller. When approved, messages detailing the patching action are sent to the stations requiring the information via ATEC telemetry at 2400b/s. Algorithm execution time and message format and dissemination is discussed in subsequent sections. Once the patching instructions have been transmitted, the timing for the manual patching action is identical to scenario one. The outage time for each circuit is reduced by nine minutes as shown by the results of this scenario.

Table 2-6 SCENARIO TWO - SINGLE LINK FAILURE

Station Legend
A - LKF
B - FEL
C - DON
D - RMN
E - KLN

NO	TIME (min.)	STATION	
1	t = 0	A	Air Force Station A receives station alarms indicating loss of receive signal from Station B. ATEC also receives alarms; begins isolation/correlation.
2	1	A	Radio maintenance advises Technical Control that the 192 channel receive from B is out.
3	3	-	ATEC assigns fault to LKF; issues fault report to area. Automated altroute search initiated.
4	4	A	Radio maintenance detects problem to be a major failure and advises Technical Control that the estimated time to repair is 3 hours.
5	5	A	Station reports that estimated time to repair (ETR) is 3 hrs. Message forwarded by ATEC to AREA.
6	6	-	ATEC and AREA Syscon data bases updated to reflect ETR and altroute plan is displayed to Syscon Controller.
7	7	-	Altroute plan approved by Controller.
8	8	All	Patching instructions arrive at affected stations. Hard copy obtained at the Communication Interface Subsystem (CIS).

Scenario Two

Single Link Failure (continued)

NO	TIME	STATION	
9	12.4	A & C	Technical Control patches two di-groups, one on space and one preempts a digroup carrying 24 ckts all with priorities less than RP2 except for two RP1 circuits which will be restored individually. There are two digroup patches restore 40 RP1 circuits (primarily VON IST's).
10	14.4	C & B	Technical Control completes patching the two di-gr ups between C & B using the two spare di-gr ups.
11	14.4	E	Station F concurrently coordinates and restores his high priority circuits.
12	47.4	A,B&C	30 RP1 circuits remain to be altrouted and are coordinated and patched individually according to priority. Spare and RPO circuits are used for this restoration.
13	74.9	A,B&C	14 RP2 circuits are rerouted next. RPO circuits are preempted to reroute the RP2 circuits. 11 RP3 circuits are rerouted next. RPO circuits are preempted to reroute the RP3 circuits.
14	76.9	A	The reporting function continues concurrently with the restoration. The last circuit restored is reported at the time indicated.

2.8.2.3 Scenario Three. Automated Altroute Search Automated Restoration-- This scenario examines the same single link failure circumstance wherein both the altroute search and the patching are accomplished automatically. This almost totally automates the process with the exception of fault confirmation times reporting times and display viewing for approval. The scenario (see Table 2-7) is effectively identical to scenario two through item 7. Three minutes later reconfiguration is complete. A subsequent section will discuss the timing in items 8 and 9 as well as message formats. All circuits and di-groups are restored in this scenario before the first circuit was restored in scenario two. Total circuit outage across all priority levels is less than ten minutes each.

2.8.2.4 Message Formats-- The automated altroute search algorithm will generate patching instructions to be implemented manually in scenario two and switching commands for execution in the CRU in scenario three.

In scenario two, the patching instructions will be transmitted to each station where patching action is required. The number of patch messages is multiplied by the number of stations involved in the restoration. The reroute of 10 circuits, for example, involving 3 stations will result in 30 messages.

Table 2-7 SCENARIO THREE - SINGLE LINK FAILURE

Station Legend
A - LKF
B - FEL
C - DON
D - RMN
E - KLN

NO	TIME (min.)	STATION	
1	t = 0	A	Air Force Station A receives station alarms indicating loss of receive signal from Station B. ATEC also receives alarms: begins isolation/correlation.
2	1	A	Radio maintenance advises Technical Control that the 192 channel receive from B is out.
3	3	-	ATEC assigns fault to LKF; issues fault report to AREA. Automated altroute search initiated.
4	4	A	Radio maintenance detects problem to be a maj r failure and advises Technical Control that the estimated time to repair is 3 hours.
5	5	A	Station reports that Estimated Time to Repair (ETR) is 3 hours. Message forwarded by ATEC to AREA.
6	6	-	ATEC and AREA Syscon data bases updated to reflect ETR and altroute plan is displayed to Syscon Controller.
7	7	-	Altroute plan approved by Controller. Messages generated for transmittal to stations.
8	8	All	Switching instructions received by CIS.
9	10	All	Channel Reconfiguration Unit has received and executed all switching instructions at all sites.

A tentative message format for circuit and di-group patching has been developed to assess the transmission times. The format for circuit patching has been derived from DCA-EURC 310-70-5 and is shown below as it may appear on a CRT display or hard copy printout.

RESTORE

ID/LINE	PREEMPT	CCSD	PATCHING STATIONS	RP	FROM	TO
01A 110		CCSDXXXX		IA	LKF	FEL
01A 111	44XYZ5/005	SPARE	LKF-DON	XX		
01A 112	44XXZ4/010	CCSDYYYY	DON-FEL	00		

Obviously the header information need not be sent, therefore, the message as transmitted is assumed to consist of the data on the three information lines. This examples corresponds to the scenarios described previously. The actual data message is assumed to be packed so that the first line will have 6 + 8 + 2 + 3 + 3 = 22 characters and the second and all following lines will have 6 + 8 + 8 + 6 + 2 = 30 characters. The number of lines after the first is a function of the patches which must be made. In the example, the circuit is first patched to DON then the FEL or two patches yielding two second lines. The message size for each patch is 22 + 30 + 30 = 82 plus header and trailer. The header and trailer for ATEC protocol is 23 characters; therefore, the total message length is 105 characters.

A similar format was derived for di-group patching.

RESTORE

ID/LINE	PREEMPT	DIGROUP	PATCHING STATIONS
02B 100		44JMXX	
02B 101	M0067A03	44ZZXX	LKF-DON
02B 102	M00XXB07	44YYXX	DON-FEL

The actual message length is estimated to be: first line 12 characters and second line and on - 26 characters yielding at total of 64 characters plus overhead for the example shown or 87 characters total when the overhead is added.

Two di-groups were patched in scenario two. The digroup patching messages will be generated first and sent to the three stations (LKF, DON, and FEL). A total of six messages consisting of 87 characters each will be sent from the originating system control element. Assuming 8 bits per character the total number of bits to be transmitted on a 2400 bits per second line is 4176 and the

transmission time is less than 2 seconds. Transmission, propagation and queue waiting times are negligible (Reference 13) when compared to the human interface at the station where the operator is told a message is waiting, the message is called-up and displayed then printed before action can be taken. One minute was allotted to this entire process. Once the station has the first patching instructions (in this case the instructions for restoring two digroups) the timing on the remaining messages is unimportant because they will have been received long before the first patch action is completed. The remaining messages contain instructions for patching 55 circuits a total of 165 messages in all. At 105 characters per message, at total of 138,600 bits are transmitted in less than one minute (58 seconds).

In scenario three, commands to the CRU are generated instead of patching instructions. The following assumptions are made relative these messages:

a. Commands are sent at 2400 b/s to the CIS and at 150 b/s between the CIS and the CRU

b. Message lengths are limited to 80 characters at the CRU (text and overhead)

c. The command message format is assumed to be a two character function code followed by a pair of channel descignators,each consisting of a 3 character port(digroup) code, a two character channel code and if appropriate a two character subchannel code,separated by a colon. The command message is like the following example:

   AS0050705:0272312

   with the packed message length equal to 17 characters.

d. More than one command can be sent in a single message as long as commands plus overhead are less than 80 characters. This means that 3 command (51 characters + 23 overhead characters) is the most in any message and the length is 74 characters.

e. The CRU command execution time is assumed to be 100 milliseconds per circuit. The CRU command to switch a digroup is equivalent to switching 24 circuits and therefore takes (24 x 100ms) 2.4 seconds.

With the above assumptions, the message load at the system control element is, based on 2 digroups and 55 circuit switching

commands, 57 commands. These commands are packed into 19 messages and sent to 3 stations for a total of 57 messages containing 4218 characters or 33744 bits. These messages, transmitted at 2400 bits/sec, take 14 sec,. to transmit. Fach station will receive 19 commands for the CRU and are transmitted to the CRU at 150 bits/sec. in the worst case situation. A total of 11248 bits transmitted at 150 bits/sec takes 75 seconds. Fach command is essentially executed as it is received by the CRU, therefore, in less than two minutes after the commands are received at the station CIS all CRU switching is complete. Three minutes were allotted for the entire message transmission, waiting, and command execution process.

2.8.2.5 Altroute Algorithm Execution-- The time between the initiation of the altroute search algorithm and the display of the altroute plan in both scenarios two and three is three minutes. The actual execution time of the algorithm has not been precisely estimated. The software is due to be sized in a later task in the program. Regardless of the software size it is felt that the algorithm will be disc access limited rather than program execution time limited. Special effort has been made to optimize the data base to keep the disc accesses to a minimum. A discussion on the data base organization is presented in Appendix A. Based on this data base organization and the restoration scenarios presented a rough estimate for the execution time of the altroute search algorithm is between 15 and 30 seconds. This appears to be a reasonable estimate based on the assumptions that the disc access time is 50 milliseconds and an average higher order language (HOL) instruction time is 25 microseconds. (1 HOL instruction = 5 assembly level instruction; average execution time for assembly level instruction = 5 microseconds.) Section 2.10 discusses improvements to this data base organization which will reduce these execution times.

2.8.2.6 Man-Machine Interface-Role of ATEC-- The assumption is that ATEC is fully deployed and that faults and alarms are reported to the node and sector elements in real time to fault isolation algorithms as required and finally to a node or sector controller for disposition. The controller views the fault report and supporting data and makes an assignment of the fault to the station level where the trouble is found to originate. The fault assignment/report is again viewed and printed in hard copy at the station level for action to resolve the problem. The action maybe further fault isolation or simply confirmation that the fault actually exists together with and estimated time to repair. The fault confirmation and "estimated time to repair" report to the node or sector becomes the input for deciding on repair or altroute. The man-machine interface is mentioned here because it is assumed that these manual interfaces will still be very much of the "way of doing business". The assignment, viewing, printing, confirming, and reporting actions are by far

the slowest timing elements and are by far the controlling times for most of the significant events in the scenarios.

2.8.3 Results

In order for the previous analysis to be useful in the benefits analysis, equations were developed which contain the restoration timing elements as derived from the scenario evaluation. These equations can be applied to other parts of the DCS where the circuit priority distribution maybe different or connectivity more limited requiring more stations to be involved in the restoration. The basis content of derived equations consist of the following elements:

- time for fault isolation and decision to altroute
- Altroute searching times
- Coordination time associated with restoration
- Reconfiguration and patch action timing
- User coordination and report timing.

The sum of these elements equals the total restoration response time. The equations may also be used to determine the specific outage time for the nth circuit being restored which is the type of timing data required in the benefits analysis.

The equations derived are given below for the three degrees of automation.

2.8.3.1 Manual Restoration--

RTMM = F + AM + C + PM + R

where RTMM = Restoration response time for manual altroute search and patching

F = Fault isolation time to decision that altroute is required; taken to be a constant of 5 minutes.

AM = Manual altroute search timing.

= KS/2 where S is the number of stations involved in the altrouting and K is a constant =6 minutes.

C = Coordination time of altroute strategy and plans

= JS/2 where S as above and J is a constant =2 minutes

PM = Manual patching time

= TRG + TRC where
TRG is the time to restore groups given by NG (GP + PT) + S-2) (GP + PT) where NG is the number of stations involved, GP and PT are group patching time and patch test/checkout time, 1.2 minutes and 1.0 minute respectively ,TRG is zero if no groups are to be restored. TRC is the time to restore circuits given by NC/NTC(CP + PT) + (S-Z) (CP + PT) where NC is the number of circuits and NTC is the number of technical controllers involved in the actual patching operation. NTC has been set to a constant of 2 for this analysis. CP, PT, and S are the same as above.

PM reduced to the following:

PM = 2.2 (NG + S-2) + 2.2 (NC/2 + S-2)

R = User coordination and reporting = 2 minutes, applied only once since it is a concurrent, on going operation.

2.8.3.2 Manual Restoration Aided by Automated Altroute Search--

RTAM = F + AA + C + PM + R

where RTAM = Restoration response time for automatic altroute search and manual patching.

F = Fault isolation same as above equal to 5 minutes.

AA = time increment to completion and display of altroute plan to syscon controller = 2 minutes.

C = time increment allotted to transmission reception, display and printout of first altroute instructions at the station level = 1 minute

PM = same as manual restoration

R = same as manual restoration

2.8.3.3 Automated altroute search and restoration--

RTAA = FI + AA + C + PA

where FI and AA are the same as above

C = is the time allotted to transmission and reception of the CRU commands at the CIS = 1 minute (rounded up to minute; only fraction of a minute required)

PA = time allotted to transmission from CIS to CRU and execution of switching command at CRU = 2 minutes

2.8.3.4 Swingate - Houtem Link Failure--

Application of the derived response time formulas to another portion of the DCS was analyzed to determine the reasonableness of the results. A link failure between Swingate and Houtem was chosen as the stress. While no detailed circuit data base is available, the assumptions regarding circuit restoration priority distribution is taken from Figure 2-21 which shows 26% of the circuits have restoration priorities higher than RP00. In addition, the projected (DEB Stage III and IV) multiplex configuration is used to make assumptions regarding digroup routing. Figure 2-16 shows the connectivity in the area of the failure. The following assumptions are made:

a. The SWG-HOU link carries two mission bits streams on which there are 10 active di-groups and 6 spare di-groups

b. The di-group routing is as follows:

Route	Grps	Ckts	No. to be Restored
HIN-FEL	2	48	13
HIN-SCH	2	48	13
HIN-SHP	2	48	13
HIN-LKF	1	24	6
HIN-BRF	1	24	6
HIN-CHV	1	24	6
LDN-SHP	1	24	6

c. The total number of high priority circuits to be restored is 63. All restoration is accomplished by circuit patching primary from HIN. The spare digroup between HIN and BRY and from BRY to MAM cannot be employed in the manual restoration process because BRY is designed for unattended operation.

d. The average number of stations involved in the restoration of any circuit is 4.

The resultant total restoration response time for this failure based on the above assumptions is shown in Table 2-8 These values have been reviewed with former tech controllers and they agree on the reasonableness of the total but suggest that some high priority orderwire circuits may be restored prior to the time indicated in the table (before 21 minutes has elapsed). For the nth circuit, the time is reasonable.

Table 2-8 SWG-HOU Restoration Response Time

Response Time Element	RTMM	RTAM	RTAA
Fault Isolation	5	5	5
Altroute Search	12	2	2
Coordination	4	1	1
Patching/Switching	73.7	73.7	2*
Reporting	2	2	0
TOTAL (Min.)	96.7	83.7	10.

*Actual worst case time is 1 min. 28 sec.

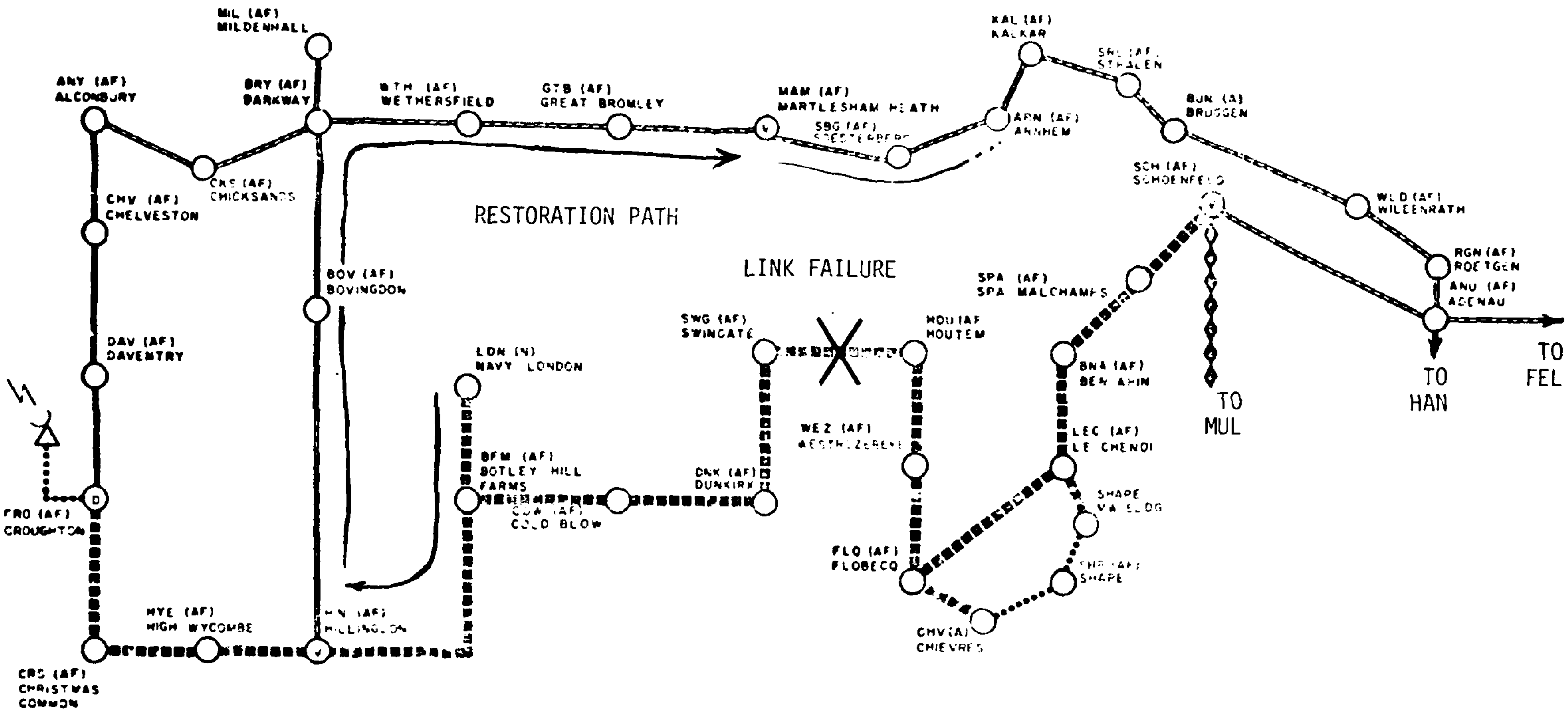


Figure 2-16. DCS Connectivity Near SWG - HOU Link Failure

2.9 BENEFITS ANALYSIS OF THE ALTROUTNG ALGORITHMS

2.9.0 Introduction to Benefits Analysis.

In order for the altrouting algorithms to benefit the DCS transmission system, the availability of circuits in the network should be improved by the automated altrouting aid. The optimum way to test the availability would be to simulate the algorithm altrouting and manual altrouting and compare their performace on selected altrouting problems. Since this is not possible at this time, another analysis must be found.

The analysis to be carried out will develop the availability in terms of a rather generalized view of the problem. The analysis tools will be given paramenters which will allow variations from one circuit to another to be seen.

The first analysis tool to be developed will treat each circuit's status as a state in a finite state markov chain. The states will represent conditions of the circuit's service, route and equipment. The transition probabilities given between states will not only lend some generality to the analysis, but will allow for differences in circuits to be made.

The availability of the circuit analyzed by the state technique will be the probability of the circuit being in any of the states where the service is available, no matter what the route. The overall circuit availability will then depend upon whether the circuit can or cannot reach the altroute states. Since we are dealing with generalized circuits in this analysis, we can only hope to estimate the probability that a circuit will be able to be altrouted. With this knowledge, a weighted sum of the availability of the circuit with and without the a priori altroute availability assumption can be made to represent the expected circuit availability. Varying the state

transition probabilities and altroute probabilities will provide ways of differentiating circuits from one another (due to such factors as geographic location, RP distribution of the minimum capacity. cut-set of the network, RP's of circuit, traveling with the circuit, etc.

2.9.1 Tools of the Benefits Analysis

2.9.1.1 Circuit Reliability Analysis--

Assumptions--The analysis of circuit reluability is carried out with the use of some standard reliability analysis tools. The main assumption to make these computations tractable is the constant conditional failure rate assumption. The implications of this assumption is discussed in this section.

The state of a circuit will be defined to indicate its condition regarding service availability and repair or altroute efforts. The probability of leaving one state for another at any time is needed in order to arrive at the times spent in each state. In particular, we are interested in any state where service is available to the user. Thus, the first thing to develop in this analysis is the probability of such transitions. The nature of this assumption is the key assumption in the analysis.

The probability of a state transition is the conditional probability that the transition will occur in the next dt time space, if it has not already occurred. The expression for such a probability is given in terms of the density function (f(t)) of the transition probability in order to develop some insight to it.

$$f\,(t/\ T > t) = \beta(t) = \frac{f(t)}{\int_t^{\infty} f(x)\,dx} = \frac{F'(t)}{1-F(t)}$$

The actual assumption to be made for this conditional transition probability is that it is equal to some constant.

$$\beta\,(t) = \lambda$$

Thus, the probability of the transition occurring during any time interval (dt) is the same at any absolute time value , so long as the transition has not taken place. Perhaps a more meaningful form of this assumption is made by finding the transition probabiltity density (f(t)) and the distribution (F(t)) functions. They are found to be:

$$f(t) = e^{-\lambda\, t}$$

$$F(t) = 1. - e^{-\lambda t}$$

These forms show that the constant conditional probability assumption really means that the probability of the transition increases exponentially in time with the time constant being the inverse of the constant conditional probaility. The time in this case is the time since the circuit first entered the state from which transitions are being considered.

We will assume that all state transitions that can occur will take on this form. The data available on equipment failure and

repair rates is given in this form already. other circuit transitions will need to estimate this parameter. To do this we note that the average time for a transition modeled above is simple the inverse of the constant conditional probability selected.

<u>The Derviation of Circuit Availability</u>--The circuit availability to the user is the number we seek from this analysis. To arrive at that figure, we must first define the state that the circuit may be in and what the constant transition probabilities are between states. From that point, the staedy state probability of being in each state is found. Adding up these probabilities for all states where the user has service leads to the availability of the circuit's service.

We begin by defining the circuit's states. Obviously the normal route is the most common state. The circuit can also be altrouted while equipment is being repaired or be in the altroute awaiting restoral of the normal route. The circuit may also have equipment failure with no altoute. In this analysis we shall assume that no mutilpe equipment failures can occur with any considerable probability. This is a reasonable assumption when figures of equipment failure time constants are compared to those for repair. Thus, only three types of failure state are used: one for each of the three types of equipment transmission, 1st and 2nd probability of reaching such a state is multiplied by the number of each type of equipment on a circuit route, the repair rate is the rate for one piece of equipment, since only one state of filure will occur with any probability. In each state of equipment failure, we may or may not have an altroute esrablished.

The transition probabilites for some small time (dt) during which obsevations are noted is now just the constants discussed earlier. We will call these constants "rates" for the remainder of this discussion.

The states to be considered are now formally defined:

(0) Normal circuit routing/all equipment functioning.

(1) No failed equipment, but circuit in altroute.

(2) Pre-empted circuit.

(3) Transmission equipment failure/no altroute available.

(4) First level multiplexer failure/no altroute available.

(5) Second level multiplexer failure/no altroute available.

(6) Transmission equipment failure/altroute in place.

(7) First level multiplexer failure/altroute in place.

(8) Second level multiplexer failure/altroute in place.

The state diagram of Figure 2-17 shows the state defined above and defines the transitions and their rates. This model will be used for the remanider of the analysis.

The following transition rate variables are used:

(1) μ_X : The repair rate for The type "x" equipment, where A is for transmission, B for 1st level mux, and C for 2nd level mux.

(2) λ_X : The failure rate for type "x" equipment.

(3) ρ : The altrouting rate for failed circuits.

(4) λ_ρ : The pre-emption rate for operating circuits.

(5) μ_r : The restoral rate for returning a circuit back to its original route.

(6) μ_ρ : The restoral rate for returning service to a pre-empted circuit.

The rates are fairly self-explanatory except for a few which are clarified here. The altroute rate (ρ) must take into account the fault isolation, altroute search and the altroute patching times. The restoral rate (μ_r) must allow for the recognition of equipment repair, algorithm recognition of the restoral possibility and patching of the normal route.

The equations governing the circuit's state transitions are derived by examining the states that can reach a particular state. The probability of that state being occupied at (t+dt) is the sum of the probabilities of reaching that state from all other states at (T+DT). This probability for each state with access to that state is the probability of being it that starting state at t, times the rate of the transition, times the time interval for investigation (dt). Collecting this set of equations for all states in the model leads to the set of model state equations. Dividing through by dt and taking the limit as dt goes to zero gives the diferential equations of state given in Figure 2-18.

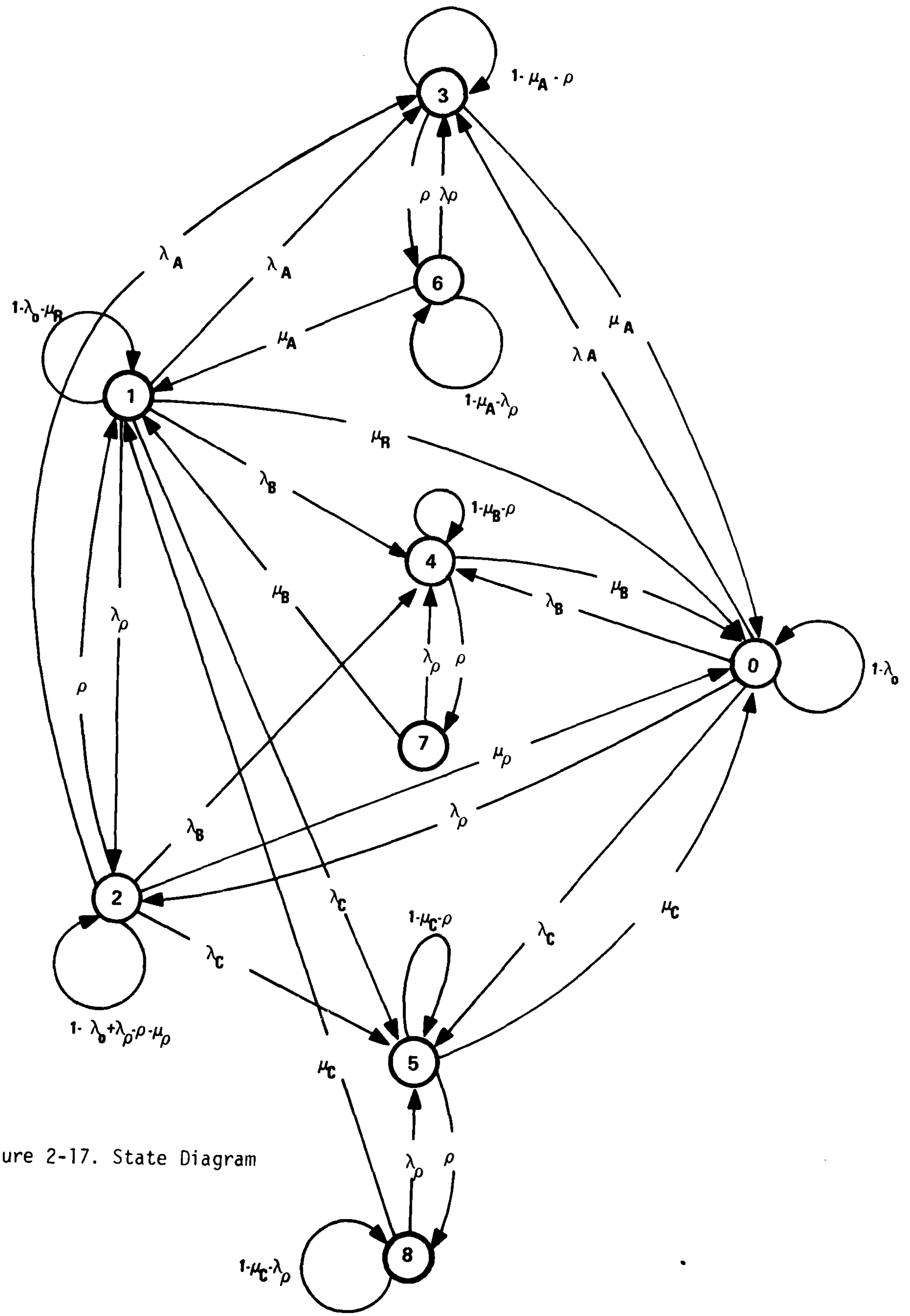


Figure 2-17. State Diagram

$$
\begin{bmatrix} \dot{P}_0 \\ \dot{P}_1 \\ \dot{P}_2 \\ \dot{P}_3 \\ \dot{P}_4 \\ \dot{P}_5 \\ \dot{P}_6 \\ \dot{P}_7 \\ \dot{P}_8 \end{bmatrix}
=
\begin{bmatrix}
-\lambda_0 & \mu_R & \mu_\rho & \mu_A & \mu_B & \mu_C & 0 & 0 & 0 \\
0 & -\lambda_0-\mu_R & \rho & 0 & 0 & 0 & \mu_A & \mu_B & \mu_C \\
\lambda_\rho & \lambda_\rho & -\lambda_\rho-\rho+\lambda_\rho-\mu_\rho & 0 & 0 & 0 & 0 & 0 & 0 \\
\lambda_A & \lambda_A & \lambda_A & -\mu_A-\rho & 0 & 0 & \lambda_\rho & 0 & 0 \\
\lambda_B & \lambda_B & \lambda_B & 0 & -\mu_B-\rho & 0 & 0 & \lambda_\rho & 0 \\
\lambda_C & \lambda_C & \lambda_C & 0 & 0 & -\mu_C-\rho & 0 & 0 & \lambda_\rho \\
0 & 0 & 0 & \rho & 0 & 0 & -\mu_A-\lambda_\rho & 0 & 0 \\
0 & 0 & 0 & 0 & \rho & 0 & 0 & -\mu_B-\lambda_\rho & 0 \\
0 & 0 & 0 & 0 & 0 & \rho & 0 & 0 & -\mu_C-\lambda_\rho
\end{bmatrix}
\cdot
\begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \\ P_6 \\ P_7 \\ P_8 \end{bmatrix}
$$

$\lambda_0 = \lambda_A + \lambda_B + \lambda_C + \lambda_\rho$

Figure 2-18. State Equations

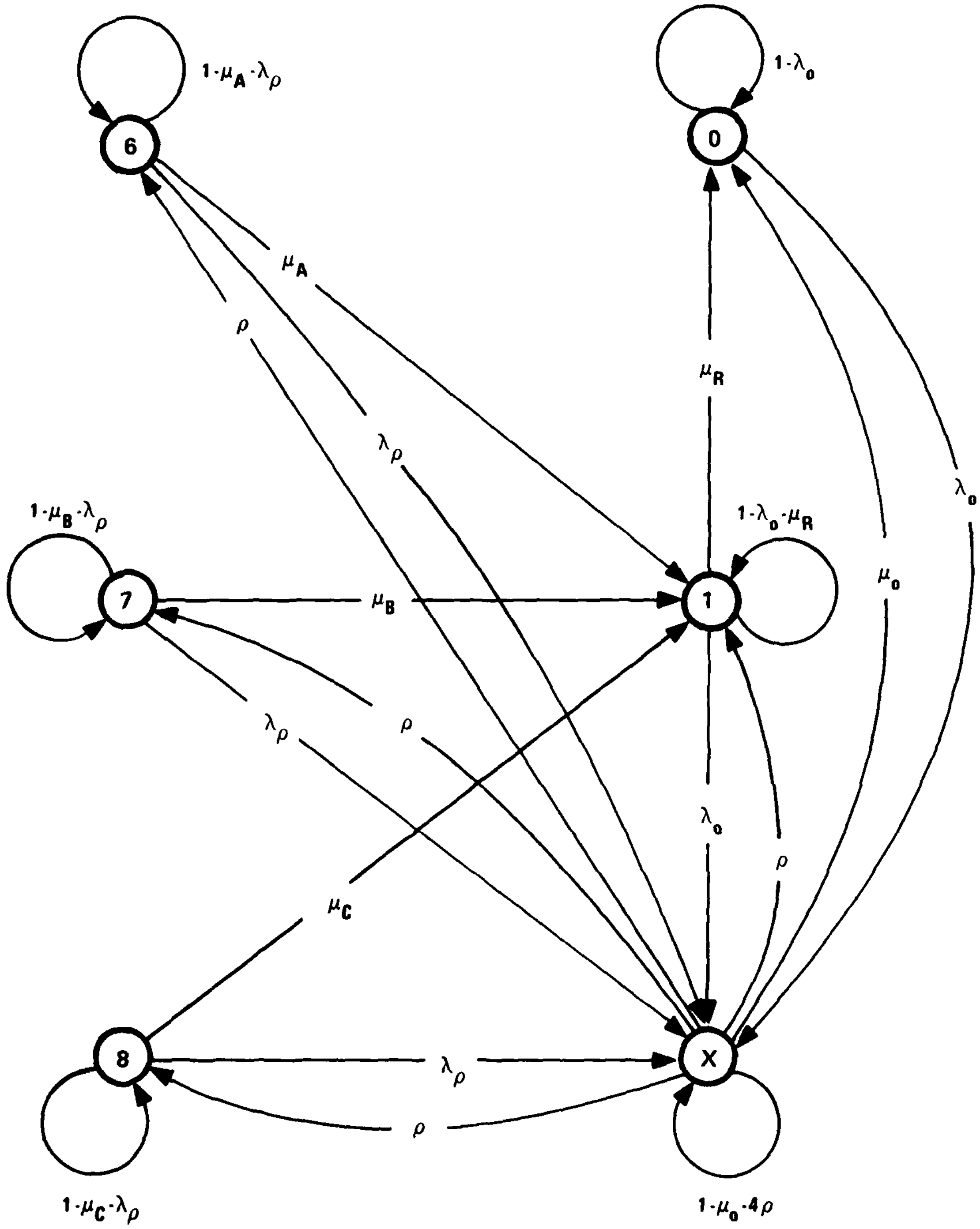


Figure 2-19. Simplified State Diagram

$$
\begin{bmatrix} \dot{P}_0 \\ \dot{P}_1 \\ \dot{P}_x \\ \dot{P}_6 \\ \dot{P}_7 \\ \dot{P}_8 \end{bmatrix}
=
\begin{bmatrix}
-\lambda_0 & \mu_R & \mu_0 & 0 & 0 & 0 \\
0 & -\lambda_0 - \mu_R & \rho & \mu_A & \mu_B & \mu_C \\
\lambda_0 & \lambda_0 & -4_\rho - \mu_0 & \lambda_\rho & \lambda_\rho & \lambda_\rho \\
0 & 0 & \rho & -\mu_A - \lambda_\rho & 0 & 0 \\
0 & 0 & \rho & 0 & -\mu_B - \lambda_\rho & 0 \\
0 & 0 & \rho & 0 & 0 & -\mu_C - \lambda_C
\end{bmatrix}
\times
\begin{bmatrix} P_0 \\ P_1 \\ P_X \\ P_6 \\ P_7 \\ P_8 \end{bmatrix}
$$

$$\mu_0 = \mu_A + \mu_B + \mu_C + \mu_\rho$$

Figure 2-20. Simplified State Equations

In this analysis, we are only interested in the time spent in the operative states (i.e. states 0, 1, 6, 7, and 8). Thus, we can combine all of the non-operative states (i.e. states 2, 3, 4, and 5) into a state x and reduce both the state diagram (Figure 2-19) and the complexity of the state equations (Figure 2-20).

The number we seek is the probability that the circuit is in one of its operative states. We will find this solution for the steady state case. This means setting all derivatives in the state equations to zero and solving for the probabilities. To make the solution meaningful, the side condition requiring that the sum of all of the probabilities add to one is also used.

The easiest solution technique is to find the probability that the circuit is in state x. THe circuit's availability to the user is then one minus the probability of state x. The solution is tedious but straight forward and is given as:

$$P_x = \frac{\lambda_o}{\lambda_o + \mu_o + \rho\{ 1 + \frac{\mu_a + \lambda_o}{\mu_a + \lambda_\rho} + \frac{\mu_b + \lambda_o}{\mu_b + \lambda_\rho} + \frac{\mu_c + \lambda_o}{\mu_c + \lambda_\rho} \}}$$

2.9.1.2 The Altrouting Probability--In order to generate an availability number for a circuit, we need to know what chance that circuit has of being altrouted. This probability depends on such factors as the RP distribution of the circuits traveling with the circuit, the distributuon of RP's of the munimum cut-set in the network seen around the failed ends of the circuit and the geographic location of the circuit.

The calculation of whether a circuit can be altrouted must next be described in order to understand the altrouting probability analysis. The distribution of circuits by RP for the circuit altroute group and the minimum cut-set seen by the ends of the failed circuits must be found. In the case of the altroute group, a distribution of the number of circuits less than a certain RP should be plotted. The number of circuits in the minimum cut-set above a certain RP should be plotted over the altroute group's distribution. The intersection of the two distributions gives the RP level that the altroute group will be altrouted to and the RP level to which the pre-empted group on the minimum cut-set will be pre-empted up to. The number of circuits where the distributions intersect also gives the number of circuits involved in pre-empting over the cut-set for the altroutes. (See Figure 2-21 for a graphic interpretation of this solution.)

The solution described above can be applied for any selected circuit altroute group and minimum cut-set seen by the group. The most difficult case in altrouting would occur when the altroute group and mimimum cut-set had the same number of circuits (i.e. a link failure forcing link altrouting over a single link cut-set). This case will be examined because it

gives a lower limit to the altroute probability of a circuit and because it demonstrates the method. The links to be considered will be statistically described. Thus, we will have capacity curves like Figure 2-21 for several levels of confidence (i.e. probabilities of having the capacity or greater). the intersection of like confidence levels of capacity gives an RP level and circuit number that is altroutable at that confidence level. Lower RP's will have lower altroute probability and higher RP's will have higher altroute probabilities found by examining the capacity curves for other confidence levels. Thus, we have a way of finding the highest altroute probability that each RP level can have. That confidence level can then be used in the calculation of an average availability for circuits of that RP level.

If the full data base were available for the network, we could find the statistical nature of circuit groups vs. RP for variuos places in the network by compiling capacities and their frequency. This is not available so that a cruder method is adopted. We select circuits by their usage types as independent random variables of a link's capacity distribution. One type is selected as dependant so as to keep total capacity limited. These variables are allowed to vary according to some assigned distribution in order to make the link's circuit type mix a variable. The RP distribution is found by using the overall network averages for RP's (Figure 2-22) (of each class in a particular circuit type) to distribute the circuits in a circuit type among the RP classes. Now by varying the circuit type's distributions, links of various circuit mixes can be developed that at least follow the network RP avareages per class.

The example to be given in detail here summarizes the results of this analysis method. The circuits of type A,D,B,C, and E were allowed to vary independantly in a noramal distribution with means equal to the overall network means for each circuit type and standard deviations of 1/3 of the mean. This made the type V circuits dependant and normal with a mean of the overall network average for type V circuits and a standard deviation of 1/6 of this mean. These standard deviations are the maximum allowed without allowing any type of circuit to have negative circuit capacity. The distribution of circuits in each RP class was taken as uniform over the RP range assigned to the RP class. The plots of the altroute group capacity and minimum cut-set capacity are given in figure 2-23. Two different confidence levels for the capacities are also plotted.

Figure 2-23 shows that there is little range of RP's at the crossing points of the capacity curves. This means that circuits are altrouted into class RP0 for nearly all confideance level of interest. Since the distribution of circuits in the RP classes was arbitrarilty chosen, there seems to be no point in carrying this solution to further detail to find specific RP levels in the RP 0 class for each confidence level.

Additional solutions for cases were widely varying circuit type means and standard deviations were allowed results in a similar conclusion as to the lowest RP level of altroutability. The reason for this result is that most circuit types have large concentrations of their circuits in RP 0 and that varying the circuits among the types does not change this RP 0 concentration. For this reason, we will assume that circuits have altroute probabilities of either 0 or 1 for future availability calculations. It seems that the uninteresting result of this analysis points out again the poor utilization of RP classes in the current network. A more gradual RP distribution would probably show more variability in the RP level for different confidence levels.

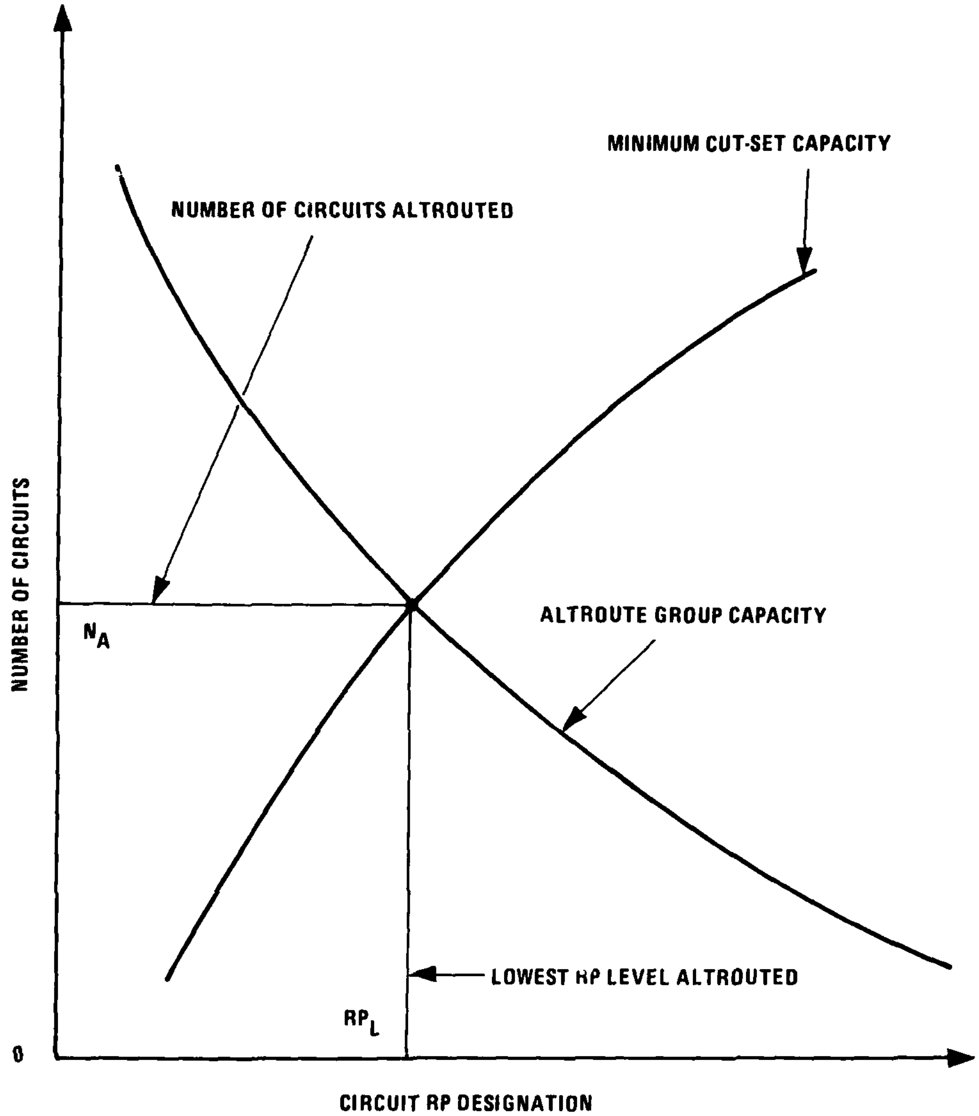


Figure 2-21. Altroutability Selection

FIGURE 2-22

CIRCUIT DISTRIBUTIONS FOR THE EUROPEAN DCS BY RP AND CIRCUIT TYPE

			Listing by RP classes				
Type Code	Usage Type	Total	1	2	3	4	0
A	Non-switched TTY	7%	2%	1%	1%	0	3%
D	" data	2%	0	0	0	0	2%
V	" voice.	60%	3%	3%	4%	0	50%
B	AUTOVON access	15%	1%	0	1%	0	13%
C	" truck	6%	3%	0	0	0	3%
E	AUTODIN access	5%	1%	1%	1%	0	2%
-	Others	5%	3%	1%	0	0	1%
	Totals=	100%	13%	6%	7%	0%	74%

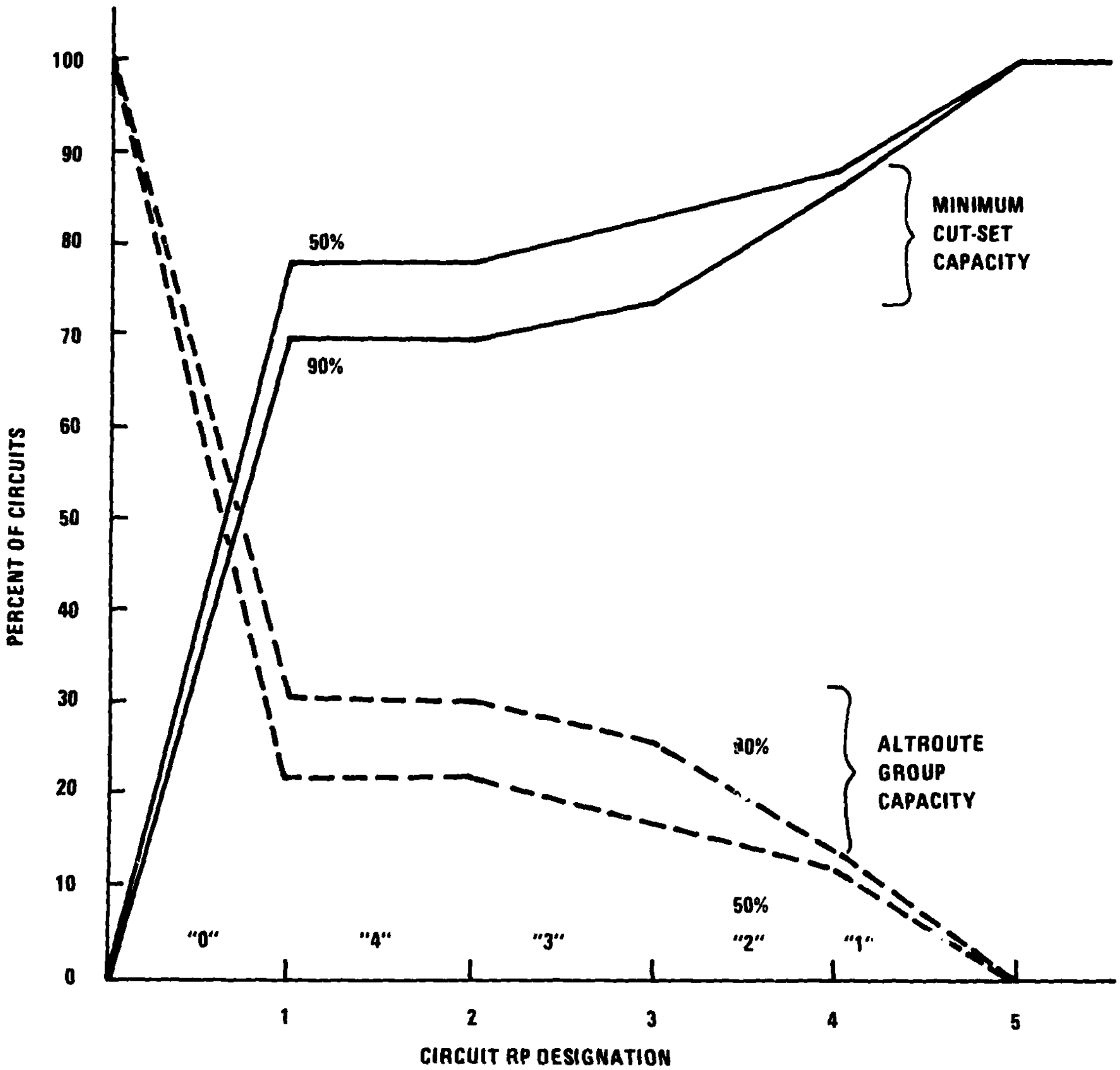


Figure 2-23. Circuit RP Designation

2.9.2 Example of Benefits Analysis

With the tools developed in the first section, we are now in a position to analyze the benefits of the altrouting algorithms.

The example to be considered is a link failure in the network which causes a large group of circuits to be altrouted. We assume that the altroutes can be found for the circuits of RP 1 to 4 with probability 1 and that RP 0 circuits cannot be altrouted (as per the discussion of section 2.9.1.2). This means that perhaps 100 circuits on the link can be altrouted. We will develop the outage probability of circuits at every level of RP in this group of 100 for the steady state circuit model developed in section 2.9.1.1.

In presenting the relative outage probabilities, several parameters will be varied:

(1) The RP of the circuit being altrouted.

(2) The number of stations of circuit level access on the normal route (this number gives us an estimate of the equipment used for the normal route.).

(3) The number of stations involved in creating the altroute (this gives an estimate of the coordination and patching effort required.).

(4) The method of altroute searching and altroute patching - either manual or automatic.

The model of the circuits used in this example is designed to appear like the average circuit route in the network. The number of links per trunk of the circuits is taken as 3 (a number used in describing the typical digital DCS trunk). The number of links per station along the altroute which must be searched for an altroute segment is taken as 1.25 (a number found as the average in the European backbone used in this report.). The number of trunks acessible per link at the search stations is taken as 3.9 (again, an average over the European backbone.).

The times used to estimate the average transition rates in the circuit state model are derived from the response time analysis given in this report. The run time of the automated algorithms for altroute searching is assumed to be disc access time limited. The disc access time assumed here is 50 msecs.. The equipment failure rates and repair rates are derived from the data given in reference 10. The failure rate of any type of equipment used on the normal route is the failure rate for one such piece of equipment times the number of such pieces used. The average circuit route assumed earlier allows us to estimate this number for radios, 1st and 2nd level multiplexers. The repair rate of each equipment type is just the repair rate of one piece of

equipment - we have alreadt assumed that multiple failures are rare and not to be considered here.

The rates of circuit pre-emption and restoral that make up the failure sum and repair sum seen in the outage expression require some assumptions. We have assumed here that the pre-emption rate is small compared to the sum of the equipment failure rates. This is probably very true for the RP 1 through RP 4 circuits being handled here under typical operating conditions. Catastrophic situations may alter this assumption. We also assume that the average time to remove a pre-emption on any circuit is the sum of the average equipment repair time (this gives us some idea of the time needed to remove the fault that caused the pre-emption as part of another circuit's altroute) and the average altroute time for the circuit if it were altrouted. This second term will reflect the patching and coordination times to remove the altroute of the pre-empting circuit if that pre-empting circuit has nearly the same RP.

With the above assumptions in mind, Figure 2-24 shows the outage probability for two circuits. One circuit has five trunks normally and is altrouted over one new trunk in one case and four in another.

The second case is a single trunk circuit altroute altrouted over one trunk and then again for the case of a much different route using four trunks.

2.9.3 Summary of the Benefits Analysis

The example of Figure 2-24 shows that the automated search routine makes almost a 50% reduction in outage probability for the most important circuits being altrouted compared to the manual search case. The benefit is in the shorter altroute search and coordination time involved. Note also that this considerable outage reduction occurs even with the relatively rapid manual altroute search and coordination times assumed. With less efficient tech controllers, the benefit can be expected to be even greater. The benefit decreases with the lower RP circuits because the altroute time for those circuits tends to be dominated by the manual patching time. Using the automated patching of a CRU network makes great improvements on this time and results in a much improved availability.

The benefit of the automated search algorithms must also be considered in situations that are extraordinary. The critical circuit altrouted during a severe failure situation in the mission oriented communications network of the DCS is often more important than the average day-to-day service in the network. If the automated algorithms can make the altroute present sooner than the tech controller efforts, then perhaps a crucial situation is aided. This rare situation should be considered when the full benefits of the automated search control function is

considered. The senarios given in the section on response times points out such cases.

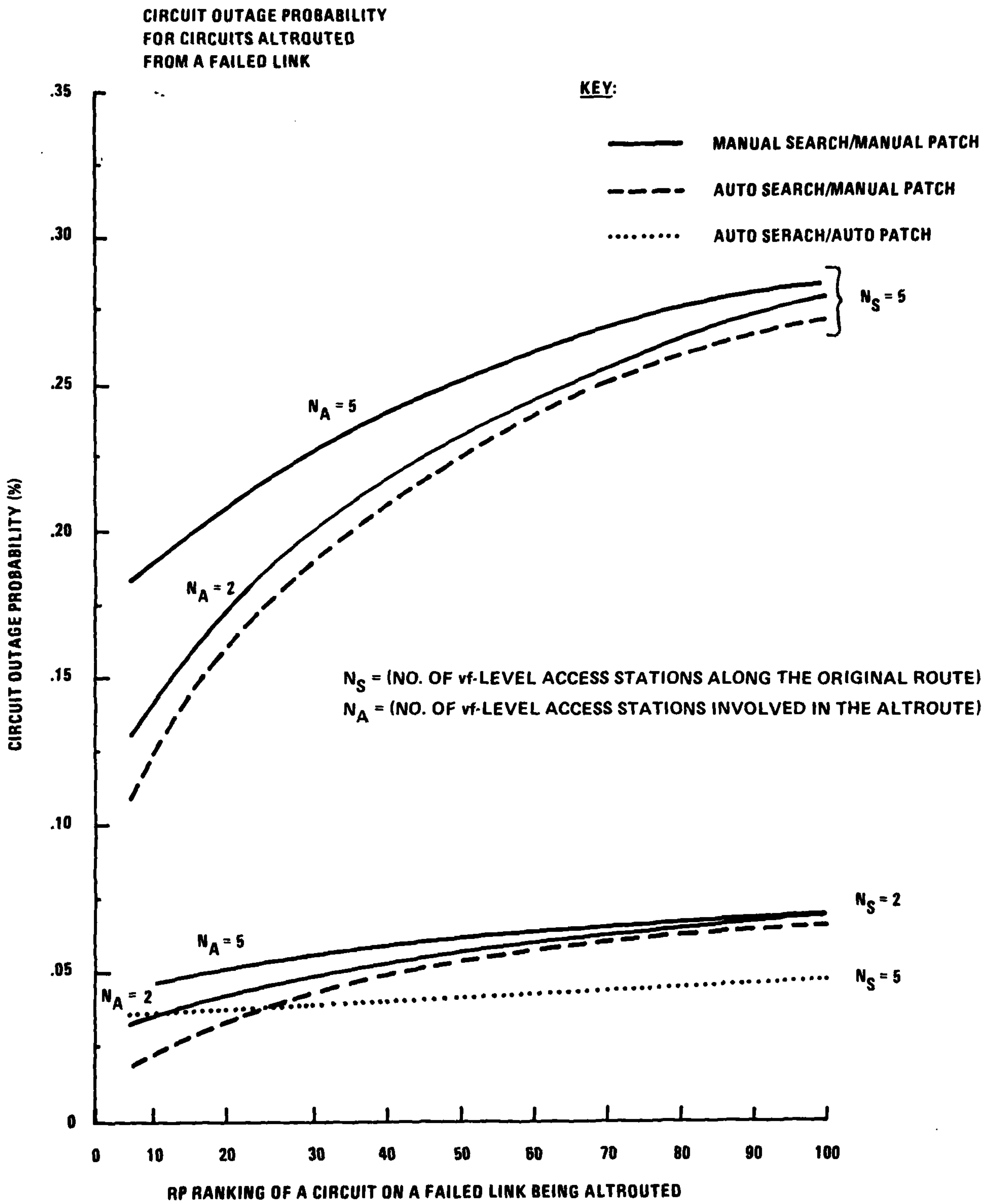

CIRCUIT OUTAGE PROBABILITY
FOR CIRCUITS ALTROUTED
FROM A FAILED LINK
KEY:
MANUAL SEARCH/MANUAL PATCH
AUTO SEARCH/MANUAL PATCH
AUTO SERACH/AUTO PATCH
CIRCUIT OUTAGE PROBABILITY (%)
.35
.30
.25
.20
.15
.10
.05
0
$N_S = 5$
$N_A = 5$
$N_A = 2$
N_S = (NO. OF vf-LEVEL ACCESS STATIONS ALONG THE ORIGINAL ROUTE)
N_A = (NO. OF vf-LEVEL ACCESS STATIONS INVOLVED IN THE ALTROUTE)
$N_S = 2$
$N_A = 5$
$N_S = 5$
$N_A = 2$
0
10
20
30
40
50
60
70
80
90
100
RP RANKING OF A CIRCUIT ON A FAILED LINK BEING ALTROUTED

Figure 2-24

2.10 DATA BASE CONSIDERATIONS

2.10.1 Introduction

Technical Report No. 2 (Ref. 2) discusses implementation of a connectivity data base (DB) that supports manual alternate route selection by a trained operator. Mainly the DB supports displays and queries from a technical control operator within the ACOC that enable him to select an alternate route. The operator is responsible for final processing of data presented to him to reach the optimum altroute. Without attempting to define the basis of optimum, this implies that the content of the DB includes all of the data recommended for use at the ACOC for real time control functions and that the relationships between system elements necessary for stress isolation, impact summaries, and available resources identification is available to the operator. The intent has been to define the data base as if it is a part of one global data base. Therefore, any system element may be traced to the responsible area. For example, each circuit file entry identifies the station which is the facility control office. The station file in turn identifies the node, sector, and ACOC responsible for that station.

The files contained in this connectivity data base are defined in Reference 2. Based on that content, an initial size estimate of about 3.6 megabytes is given.

In this report an automatic algorithm for choosing or recommending alternate routes is proposed. This algorithm is based upon the network connectivity data base just described. An overview summary of the processing involved with the implementation of this algorithm suggests that the computer resources required could become unnecessarily large. This outcome therefore suggests that perhaps another look at the DB might be in order.

This does not represent a step backward in the design considerations for such a system. In fact, it is difficult to determine which should be first. It seems perfectly logical that the algorithm and DB should be developed together. The DB must support the processing required by the algorithm. At the same time, the algorithm must be able to take advantage of features of the DB and its support systems. To a certain extent the processing required of the algorithm can be simplified if the proper data structures and inter-record relationships are maintained. In addition, the algorithm should be designed to take advantage of the capabilities of the data manipulation language (DML) associated with the chosen DBMS.

The remainder of this section attempts to look at the processing requirements of the alternate route algorithm and describe the data structure features that can minimize the processing required by the algorithm. At the same time the data structures must

support other functional requirements of the control system that use or update the data in the data base.

2.10.2 Data Base Concepts

Before looking into the issues and considerations mentioned above, a review of some data base concepts and the definition of a notation to describe a data base structure are necessary.

In its most primitive form, a data base is a centralized collection of all data stored for one or more related applications. Current direct access hardware technology permits data for many applications (or users) to share the same storage device. This, in turn, makes it possible for two or more users (application programs) to use a common, single source of data and therefore eliminate the cost and complexity of data redundancy. Once data has been integrated, a need arises for the ability to structure data in a manner which meets the requirements of each application. For the system control application being addressed here, the primary "user" of DB information will be the alternate route determination algorithm. However, there will also be secondary users of the DB such as programs that build periodic reports and programs that may be required to update static DB content from time to time.

During the design of the DB, a property called "data independence" must be kept in mind. It is this property of data base systems and the separation of data descriptions from the restrictions and conventions of any programming language that allow centralized data base maintenance, protection, and control over the physical aspects of the data base. A data base can then be viewed as more than an ordinary collection of data for several related applications. The data base must be viewed as a generalized, common, integrated collection of application data which fulfills the data requirements of all programs which access it. In addition, the data within the data base must be structured to model the natural data relationships which exist in the application.

There are three elements of the DB system that will be reviewed briefly, these are:

(a) The physical storage structure

(b) The data and control information

(c) The logical data relationships

2.10.2.1 Physical Storage Structure--The physical storage structure of a data base can vary considerably depending on the design of the direct access storage device and the manufacturer of the machine, and the number of units required to hold the DB content. As an example, consider a single disc unit which

contains 404 cylinders (or arm positions). Each cylinder contains 19 tracks (or recording surfaces), each of which has capacity for four 3,156-byte blocks of information. Thus, the physical data base is subdivided into 404 x 19 x 4 = 30,704 contiguous blocks. Each block of information is called a page and is usually the unit of physical data transfer between the data base and main memory of the system. Pages are numbered in consecutive order beginning with the first block in the first track of the first cylinder and ending with the last block of the last track of the last cylinder. The page numbers will range from 0 through 30,703. In this manner, every page has a unique number identifier and occupies a known location within the data base. "Area" is the name usually associated with a given subdivision of the disc which makes up contiguous subset of the blocks available.

The physical aspects of the DB are discussed because the operating system (OS) which hosts the DB management system (DBMS), must provide an interface between the DBMS and the physical device. This interface is usually provided by a combination of software in the OS and hardware/firmware in the device controller. By whatever method provided this interface will be referred to as the "Container Manager". The Container Manager is almost always transparent to the user or DB application programmers. However, the designer of the DB structure (the DB administrator) may be able to take particular advantage of certain aspects of the container manager in optimizing the DB design. Therefore, knowledge of the Container Manager can be very important to a proper DB design. At this point in the design of a DB for system control applications it is too early to make these kinds of considerations because the host machine and the particular DBMS have not been chosen. This is only one of the reasons why DB design for System Control must be an iterative process.

2.10.2.2 Data Base Content--The smallest unit of named data in a data base is a data item. In addition to a name, a data item has other attributes which define its type and length. A data item may be described by CKTID PICTURE/X(11) where CKTID is the name of the data item and the X(11) picture indicates that the length of the item is 11 bytes and may contain any character in the machines's character set. An occurrence of a CKTID data item could have a value such as FMD-0031667.

A record is a collection of one or more data items. A record description consists of its name followed by the names and attributes of all data items included within the record. The record named CIRCUIT could contain the data items:

CIRCUIT RECORD CONTAINS:
CKTID PICTURE/X(11)
CKTORG PICTURE/X(12)
CKTDST PICTURE/X(12)

CKTTYP PICTURE/X(12)

where CKTID is the identification number assigned to the circuit followed by the origin of the circuit (CKTORG) and its destination (CKTDST) and type (CKTTYP). Any reference to the CIRCUIT record implies reference to all data items within the record. This description may be considered a model or template for the CIRCUIT record type wherever it appears in the data base.

An occurrence of a CIRCUIT record type exists when a value for each data item exists within the data base.

The distinction between a record type and a record occurrence is important. Note that any number of CIRCUIT record occurrences may appear in the data base, and each occurrence will contain a string of characters which are defined by the CKTID, CKTORG, CKTDST, and CKTTYP data item description.

In addition, modern data base technology requires that each record occurrence of a particular type must be distinguishable from all others of the same type. That is, the collection of items that make up a record must be unique. If the defined data items for a particular record type cannot guarantee uniqueness, then an artificial data item must be created to do so.

A file is a collection of one or more records of the same type. In general a file is an artificial concept that allows one to draw a meaningful schematic representation of the DB structure as we shall see later.

Physical placement of records within a data base is controlled by specification of one or more areas in which record occurrences may be stored. In addition, one record type may be stored close to another record type to improve execution performance of the system. Any number of record types may be specified within any given area. Unless otherwise restricted, any number of record occurrences may appear for any given record type, subject to the total physical storage space limitation of the specified area. In addition, an occurrence of any record type specified for an area may be stored in any page in the specified area.

In addition to record occurrences, the data base may also contain system information used to control access to each page, provide audit trail information, and inventory available space on each page. System information may be stored either as system generated records on directories or be stored as system generated data items appended to user defined records.

2.10.2.3 Logical Data Relationships--The concept of logical data relationships is the most significant and most complex of the DB concepts being discussed in this section. Logical relationships can and usually do exist at many levels within a DB.

The most common and familiar type of data structure exists within a single record of a type. The CIRCUIT record type is an illustration of intrarecord data structure where the CKTID, CKTORG, CKTDST, and CKTTYP data items have an implied logical relationship to each other by their appearance together within the same record. Intrarecord data relationships are largely determined by the data content of the record and the meaning imparted to it by logical procedures within the application program. Intrarecord data structure is an important and useful capability which is essential in all data base applications. This type of logical relationship is inherent to virtually all DB systems.

However, to be of any significant use, a DBMS must provide a flexible method of establishing logical relationships between records of the same and/or varying types. A logical relationship between multiple records of the same type is called "proximity" or "parallelism". A logical relationship between two or more records of different types is called a "set". The set is the fundamental building block of most DBMS that allows data structures to be defined. Figure 2-25 is a representation of a set which includes three record occurrences shown by rectangular boxes. A set must have only one record type which functions as owner of the set. When that record type is defined as owner of a set, every record of that type in the DB is the owner of a different set. In addition, sets must have at least one record type which functions as a member of the set. Figure 2-25 shows one owner record occurrence and two record occurrences which participate as members. The member records may be of the same type or different types but must always be of a type different from the owner.

To clarify this further, four basic types of set relationships are illustrated below. These Figures use the shorthand notation mentioned earlier. Each box represents a file. Files are shown connected or related with arrows. These arrows are sets.

(a) Figure 2-26(a) shows a file of type A records as owners of a set in which type B records are members. Each type A record owns such a set. Each type B record may belong to at most one set at any given time or may belong to none. Each set may contain any number of type B records but only one type A record (the owner). Figure 2-26(b) shows a specific example of the set.

(b) Figure 2-27(a) shows a file of type C records as owners of a set in which both type D and type E records participate as members. Each type C record owns such a set. Any given type D or type E record may belong to at most one set at any given time.

Figure 2-27(b) shows a specific example. The type D and type F records may be in any order.

(c) Figure 2-28(a) shows a file of type F records as owners of two separate sets. Each type F record is

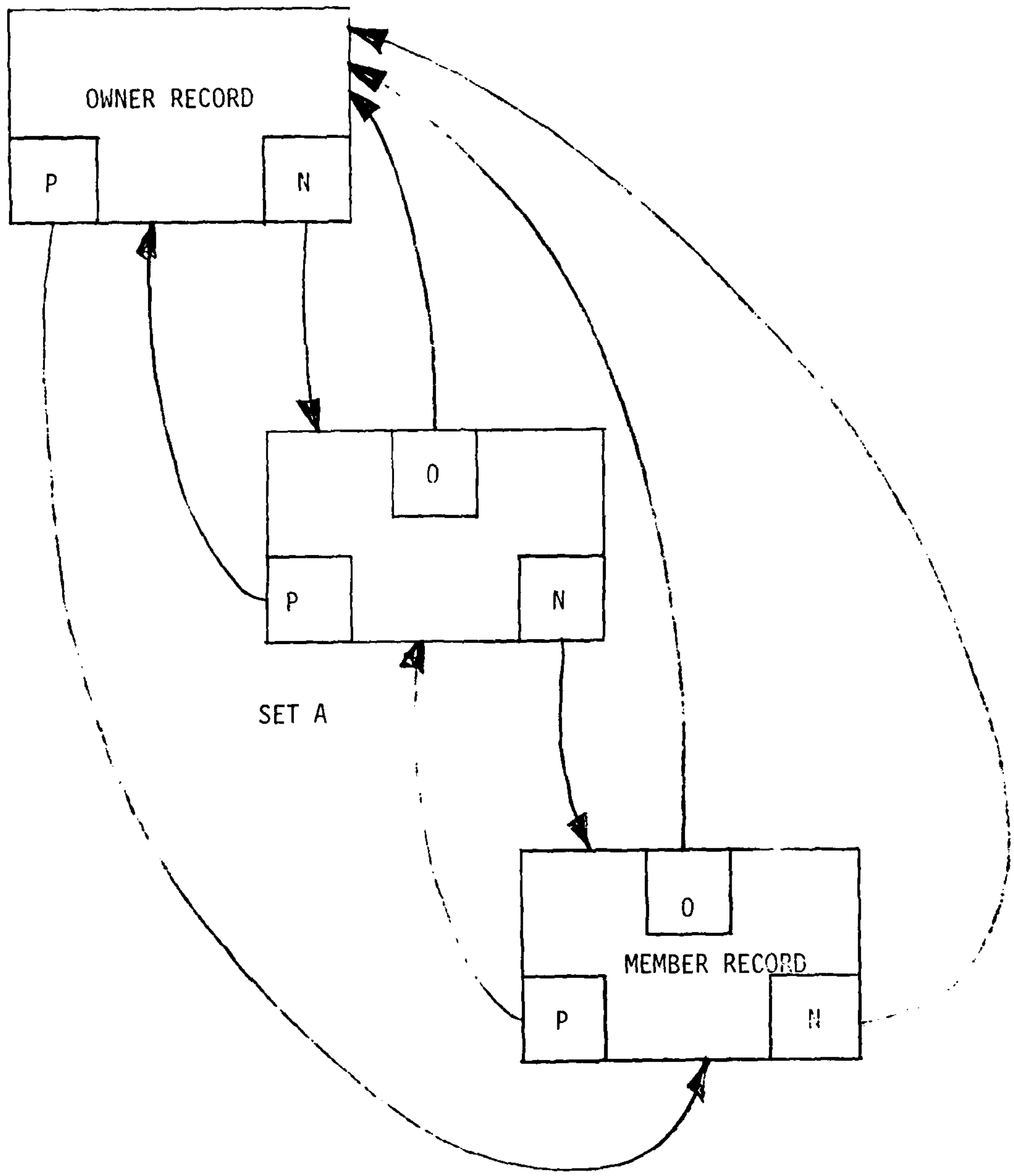
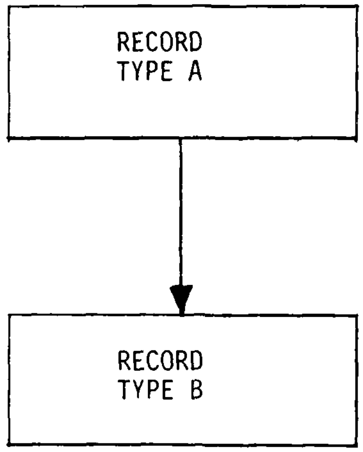
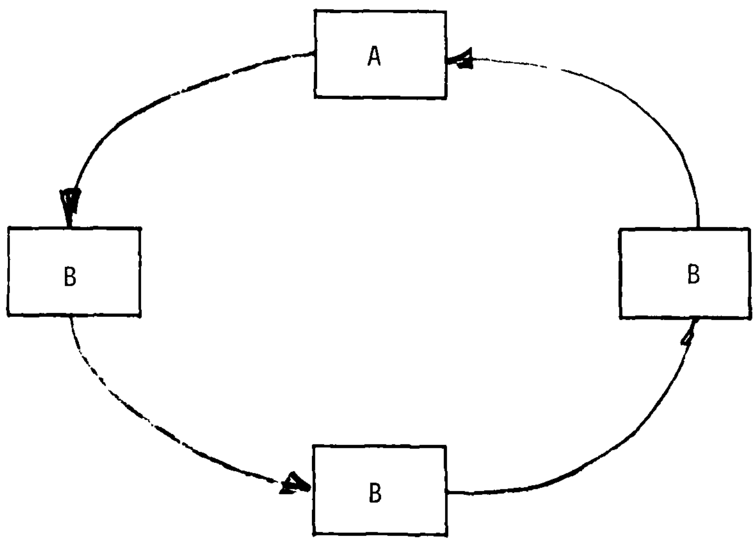


Figure 2-25. Set with NEXT(N), PRIOR(P), and OWNER(O) Pointers

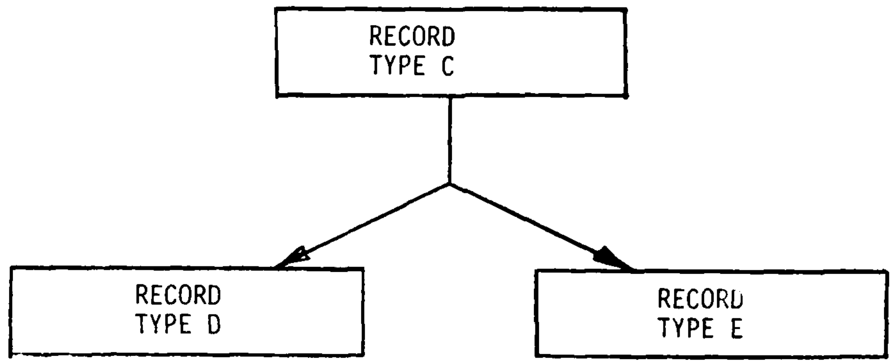


(a) Schematic Representation

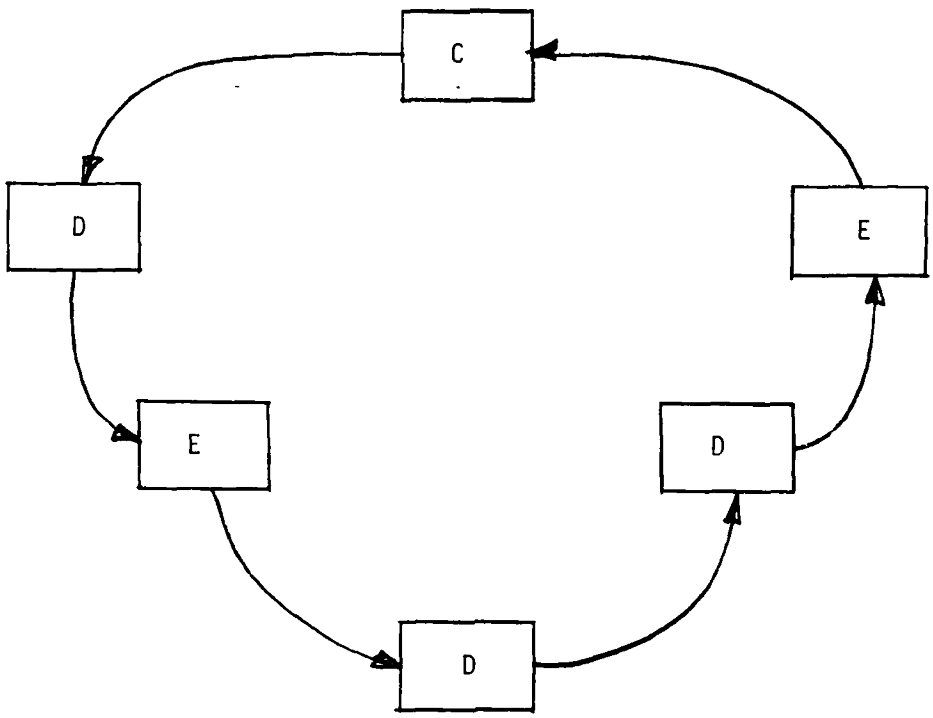


(b) Sample Configuration

Figure 2-26. Single Owner/Single Member Type Data Relationship



(a) Schematic Representation



(b) Sample Configuration

Figure 2-27. Single Owner/Multi Member Data Relationship

owner of one of each type set. One set has as its members, type G records and the other, type H records. Both type G and type H records may belong to sets owned by the same type F record but not to the same set. Figure 2-28(b) shows a specific example of a multi-ownership data relationship.

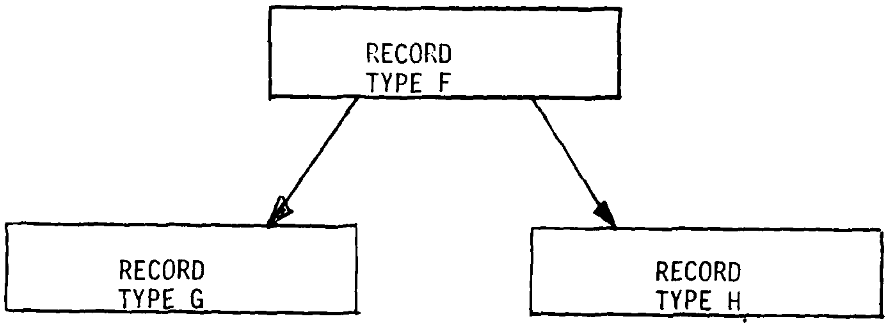
(d) Figure 2-29(a) shows a file of type L records which may be members of sets owned by type J records and type K records simultaneously. Any type L record may belong to at most one set owned by a type J record and one set owned by a type K record simultaneously. Figure 2-29(b) shows a specific example of a multi-membership data relationship.

The owner of a specific set and the members of that set are all said to be members of a "chain". It is the chain that provides for grouping of sets in the DB. There are a variety of methods of maintaining and manipulating chain contents. In general, the method is implemented by the DBMS's "Content Manager". Most DBMS systems use pointers or imbedded link structures to maintain chains. Very few DBMS use both methods, however if a DBMS supports the pointer method, the imbedded link structure is easily implemented by the application. Both methods have advantages and disadvantages. A third method maintains data relationships through the use of relational algebra--a procedure discussed later.

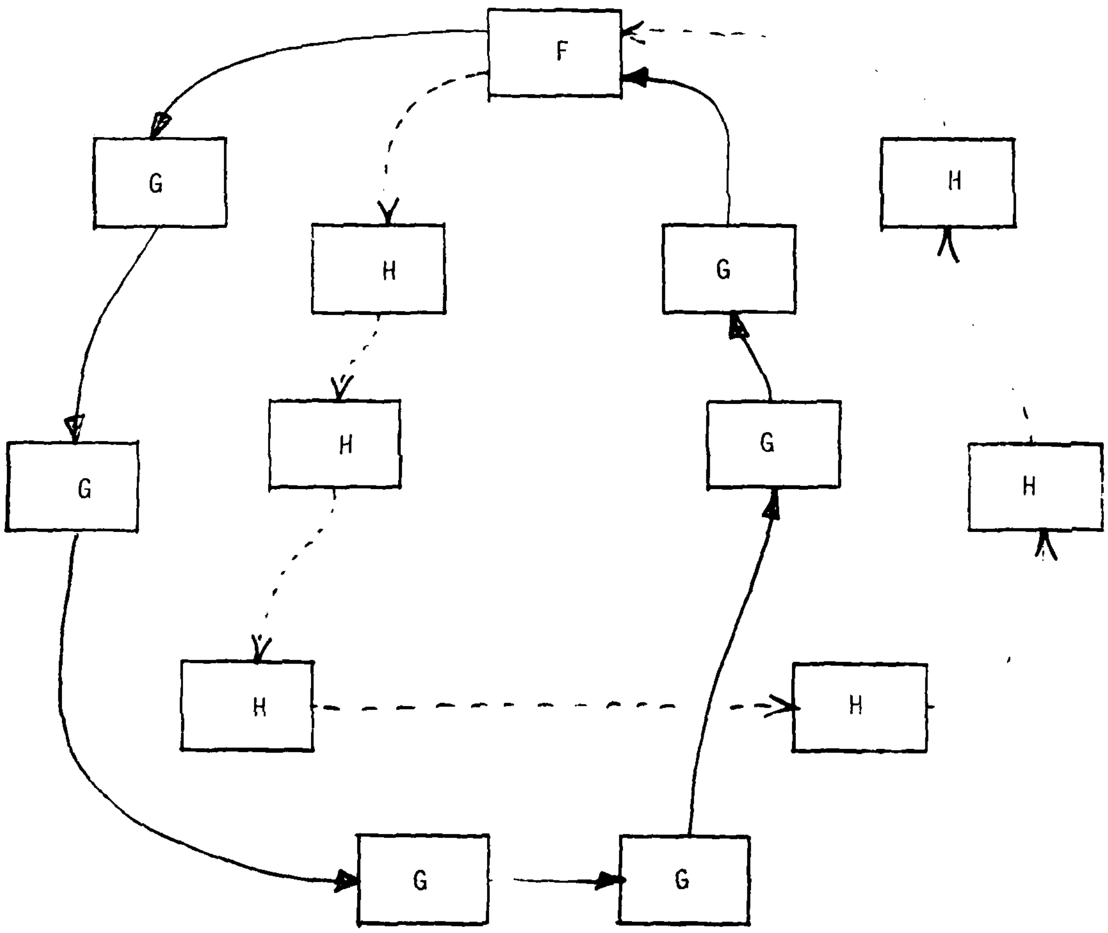
Figure 2-25 can be used to illustrate the first two methods. The pointers or imbedded links are included with the data as part of each record occurrence. The pointers or links become implied data items in the records. If pointers are used, the pointers are usually related to the physical structure of the DB container (often the DB block number where the record is stored is used). If imbedded links are used, the identifier (that part of the record which makes it unique from all other records of the same type) is imbedded in the data item. The DB Content Manager must then translate the identifier to a value it can use to find where the linked record is. The connectivity DB discussed in reference 2 discusses virtual links and direct links. These correspond respectively to the pointers and imbedded links discussed here.

By whatever method, the owner record contains a pointer marked "N" (next) which identifies the first member record occurrence (see Figure 2-25). The first member record occurrence also contains a pointer marked "N", which identifies the second member record occurrence in the set. Finally, the last record occurrence contains a pointer marked "N" which identifies the owner record. Taken together, all of the "N" pointers establish a logical chain order in the NEXT direction.

The pointers marked "P" (prior) establish a logical chain order in the PRIOR (reverse) direction. The owner contains a pointer marked "P" which identifies the last member record occurrence in

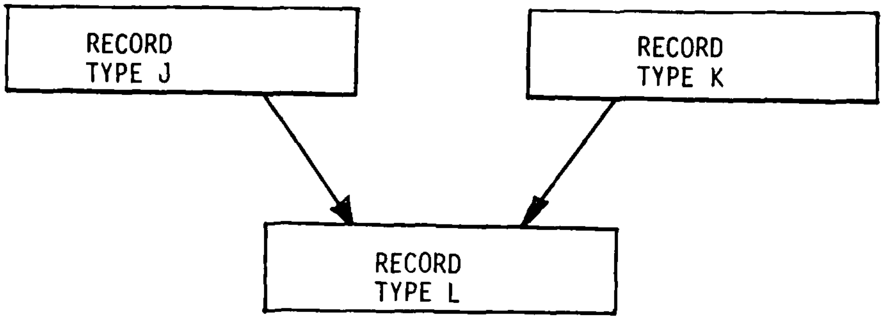


(a) Schematic Representation

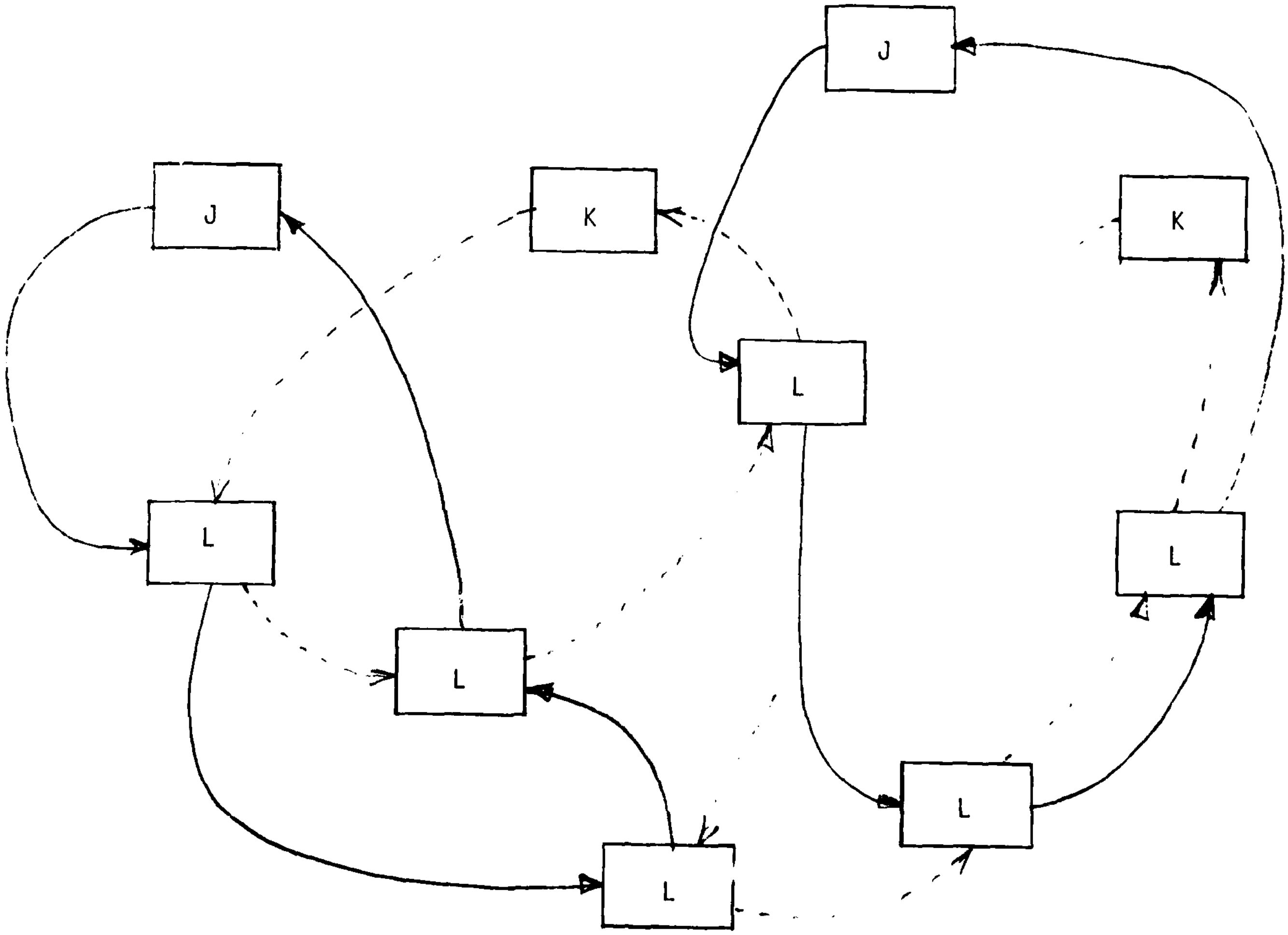


(b) Sample Configuration

Figure 2-28. Multi-Ownership Data Relationship



(a) Schematic Representation



(b) Sample Configuration

Figure 2-29. Multi-Member Data Relationship

the set. The last record contains a pointer marked "P" which identifies its logical predecessor which in turn points to the owner record occurrence. In addition each member record occurrence contains a pointer marked "O" which identifies the OWNER record occurrence. The NEXT and PRIOR chains along with OWNER pointers are considered as a model or template for all occurrences of the set named "A". Note that a data base may contain any number of owner record occurrences which in turn may have any number of member record occurrences.

In general, since all applications do not require chains linked in both forward and backward directions most DBMS's provide some mechanism for specifying which types of chain linking mechanisms are desired for a particular set. Additionally, the DBMS software (i.e., the DML interface) must automatically maintain the set link pointers. Each time a record is added or deleted, the pointers to/from that record must be updated.

2.10.3 The Data Base Content Manager

The Content Manager associated with a DBMS presents the logical interface to the DB user. The Content Manager provides an implementation of the data manipulation language or DML. The DML is like a programming language that lets the DB administrator define the DB in terms of the data items, records, files and their implied or specific relationships. This essentially specifies the schematic representation of the DB in a format that can be used by the DML to store information into the DB and retrieve it again on command. This segment of the DML is often referred to as the Data Description Language (DDL).

In addition the DML must provide logical data base access to the information contained in the DB. This capability is usually reviewed as an interface to or an extension of a host language like FORTRAN or COBOL. The DML functions can be grouped into control, retrieval, and modification categories.

2.10.3.1 Control--Control statements are used to obtain access to an area within the data base. Examples include the OPEN statement which announces the user's intention to begin processing within one or more specified areas of the data base. When access is established by the data base management system, retrieval or modification statements may be executed. The CLOSE statement announces completion of processing in the specified areas of the data base.

2.10.3.2 Retrieval--Retrieval statements are primarily concerned with locating data in the data base and making it available to a program. Most DML language specifications provide a variety of methods for access of record occurrences within a data base:

(1) Direct access of any record occurrence in the data base is possible provided that its system-assigned unique identifier is known. This type of access is independent of any set relationships associated with the record.

(2) A record type may be stored and retrieved based on the value of one or more data items contained within the records. The data base management system uses the data item value to "calculate" (CALC) the position within the data base to store each record occurrence. To retrieve a record occurrence, the user must furnish the value of the specified data item before execution of the retrieval statement.

(3) Record occurrences may be accessed through their participation in one or more sets. Once a record occurrence has been retrieved, the sets in which it participates as either owner or member provide an access path for retrieval of other associated record occurrences.

(4) Records which participate as a member in a set may be specified as ordered in either ascending or descending sequence based on one or more data items contained within the record. Nonordered sets may be automatically searched to find a record occurrence with data item values matching values supplied by the program.

(5) All occurrences of any record type may be accessed by a complete scan of an area starting with the first page and ending with the last page in the area. This method of access is independent of any other set relationships or location mode.

2.10.3.3 Modification--Modification statements result in a change to the contents of the data base. Changes include the addition of new data, modification of existing data by replacement of data item values, or deletion of data which currently exists in the data base. Modification statements are also provided which permit participation of existing record occurrences in specified sets to be established or removed.

2.10.4 Data Base Classification

For this application, the DB must be viewed as a collection of stored operational data which is used by an application system which must support multiple and user applications of the data. The on-line users represent a real time need for information in the data base. The on-line users, (i.e., the technicians and network controllers) demand immediate access to DB information concerning circuit availability, status, etc. Their interface is

via applications programs written specifically to retrieve this information and make it available.

At the same time there are application programs designed to enable the alternate route algorithms. These programs access the DB to obtain route information, end points of specific circuits segment, link cost of service information, etc.

A third type of DB interface is needed to supply high level users with information requested on an ad-hoc basis. At the sector level, where a wider view of the network is available, a query language (which is an application program designed to handle ad-hoc information requests) can be provided. All these views of the DB system must be considered in selecting a DB structure and a DBMS for this system control application. This procedure is referred to as DBMS classification.

The classification of a DBMS is based on the way in which the DBM views the relationships between files in the DB (i.e., the data structures and file organizations). The classification of DMBS's is important because each classification has distinct advantages and disadvantages which must be addressed before a system control DBMS can be selected. The best way to approach this is by example in the system control context.

Consider the set of sample data which is represented in Figure 2-30. It consists of three sets of data. Table T represents information about trunks. They are described by an identifying data items such as; number (T#), a descriptive name (TNAME), their current status (STATUS), and their endpoints (SOURCE and TERM). Table C represents information on circuits which ride on the trunks. These are described by a circuit or CCSD number (C#), a user of the circuit (USER), their current status (STATUS), a local phone contact point (PHON#) and a maintenance code (MAINT). Table TC indicates which circuits use which trunks (C# and T#), and where the connections are made (PATCH).

An essential feature of this set of sample data is that there is a many-to-many correspondence between circuits and trunks. Each trunk can host many (more than one by definition) circuits and a circuit may ride on multiple trunks to connect two end users. This relationship is highly typical of a real network environment. There is a corresponding relationship between trunks and links.

2.10.4.1 The Hierarchial Approach--The Hierarchial Approach to DB management was devised as a method for processing large amounts of data where most data processing was performed with purely sequential media. For the circuit/trunks sample data it is possible to represent the information in a manner in which trunks are superior to circuits.

This model then presents five hierarchical occurrences, one for each trunk. Each occurrence contains one trunk record and one circuit record for each circuit hosted by that trunk. Each circuit record occurrence includes appropriate patch data from Table TC. This approach is summarized in Figure 2-31.

TABLE TC

C#	T#	PATCH
C1	T1	36/08
C1	T5	27/05
C2	T1	18/97
C3	T2	72/81
C3	T4	19/27
C3	T5	45/89
C4	T4	56/40
C4	T5	39/79
C5	T3	16/19
C5	T1	05/63
C6	T3	63/51
C6	T1	95/80
C6	T5	23/29

TABLE T

T#	TNAME	STATUS	SOURCE	TERM
T1	CABLS	GRN	CITY1	CITY4
T2	VFTRK	GRN	CITY2	CITY3
T3	AMBCT	VEL	CITY2	CITY1
T4	LRPTS	GRN	CITY3	CITY4
T5	AZFDØ	RED	CITY4	CITY5

TABLE C

C#	USER	STATUS	PHON#	MAINT
C1	DCA	GRN	3607	A
C2	NADC	GRN	1245	B
C3	NAVY	GRN	7093	A
C4	ARMY	VEL	8512	B
C5	NADC	GRN	6077	C
C6	AFRCE	RED	4236	B

Figure 2-30. Sample Data Sets.

2-234

T1	CABLS	GRN	CITY1	CITY4	
C1	DCA	GRN	3607	A	36/08
C2	NADC	GRN	1245	B	18/97
C5	NADC	GRN	6077	C	05/63
C6	AFRCE	RED	4236	B	95/80

T4	CRPTS	GRN	CITY3	CITY4	
C3	NAVY	GRN	7093	A	19/27
C4	ARMY	VEL	8512	B	56/40

Figure 2-31 Hierarchical Model of Circuit/Trunk Sample Data

The unit of access (the smallest amount of data which may be transferred) in a hierarchical data model is normally the record occurrence. It is fundamental to the hierarchical view that any given record occurrence takes on its full significance only when seen in context. No record occurrence can exist without its superior. To retrieve a particular circuit occurrence, the user must state which circuit he is interested in and which trunk that circuit is on. The Hierarchical DML provides a construct to allow the user to access a "unique" record occurrence provided he supplies sufficient information to identify the entire hierarchical path involved. In addition, hierarchical DML's typically allow the use to access the "next" record in sequence or the next which satisfies some condition.

The major advantage of the hierarchical approach is that it obviously provides a very natural way of modeling a hierarchical structure from the real world. However, difficulties arise when we try to operate on such a model in a system control environment. Consider the following sample queries and the DML statements required to answer them.

```
Q1 - Find the numbers for the circuits which use trunk T4
Q2 - Find the numbers for the trunks which host circuit C6
R1 - Step 1 - get unique trunk with T#=T4
     Step 2 - get next circuit for trunk
     Step 3 - was a C# found (if not, exit)
     Step 4 - Print C#
     Step 5 - go to step 2
R2 - Step 1 - get to start of data
     Step 2 - get next trunk #
     Step 3 - was a trunk record found (if not, exit)
     Step 4 - get next circuit for this trunk with C#=C6
     Step 5 - was a C# found (if not, go to step 2)
     Step 6 - Print T#
     Step 7 - go to step 2
```

Observe that even though the two original queries are symmetric, the DML procedures required to answer them are not symmetric. This is one drawback of the hierarchical model, unnecessary complexity. The user is required to solve a problem which is introduced by the model and not intrinsic to the question being asked. Matters will rapidly become worse as new types of records are introduced and the hierarchy becomes more complex, and programs could be more complicated than they need be.

The hierarchical model for trunks and circuits also suffers from a number of anomalies with storage operations. These are the direct result of the fact that the trunk/circuit problem involves a many-to-many relationships whereas a hierarchical DB is best used to handle one-to-many data relations. Some examples of these anomalies are:

(1) Adding a new record may not be possible.

(2) Deletion of a trunk record from the DB may cause the loss of other data as well.

(3) Updating can cause problems. To change the status of a circuit from GRN to RED the DML must search the entire DB to find every occurrence of the circuit record.

In practice, various implementations of hierarchical DBMS have gotten around these problems (as far as the user is concerned at least) by special techniques for gimmicks. However, there is always a tradeoff involved. Sometimes it involves storage requirements, sometimes retrieval efficiency.

2.10.4.2 The Network Approach--The network approach to DB manipulation is typical of systems proposed by the CODASYL DB task group (DBTG) recommendations. A network is a more general structure than a hierarchy because a given record may have any number of immediate superiors as well as any number of immediate subordinates. This enables representation of many-to-many relationships in a reasonably direct manner.

Figure 2-32 is an example of part of the same sample data arranged in a Network type DB. In addition to the circuit and trunk records in the data base, a third type of record (linking record) is introduced. A linking record occurrence represents a connection as a relationship between a trunk record and a circuit record. The linking record contains information which describes the connection. All link occurrences for a given trunk are placed on a chain starting at and returning to the trunk record. Similarly all links for a given circuit are placed on a chain.

The DML for network models permit the user to traverse the various connecting chains. Now consider the queries Q1 and Q2 and the DML statements required to answer them in the network approach.

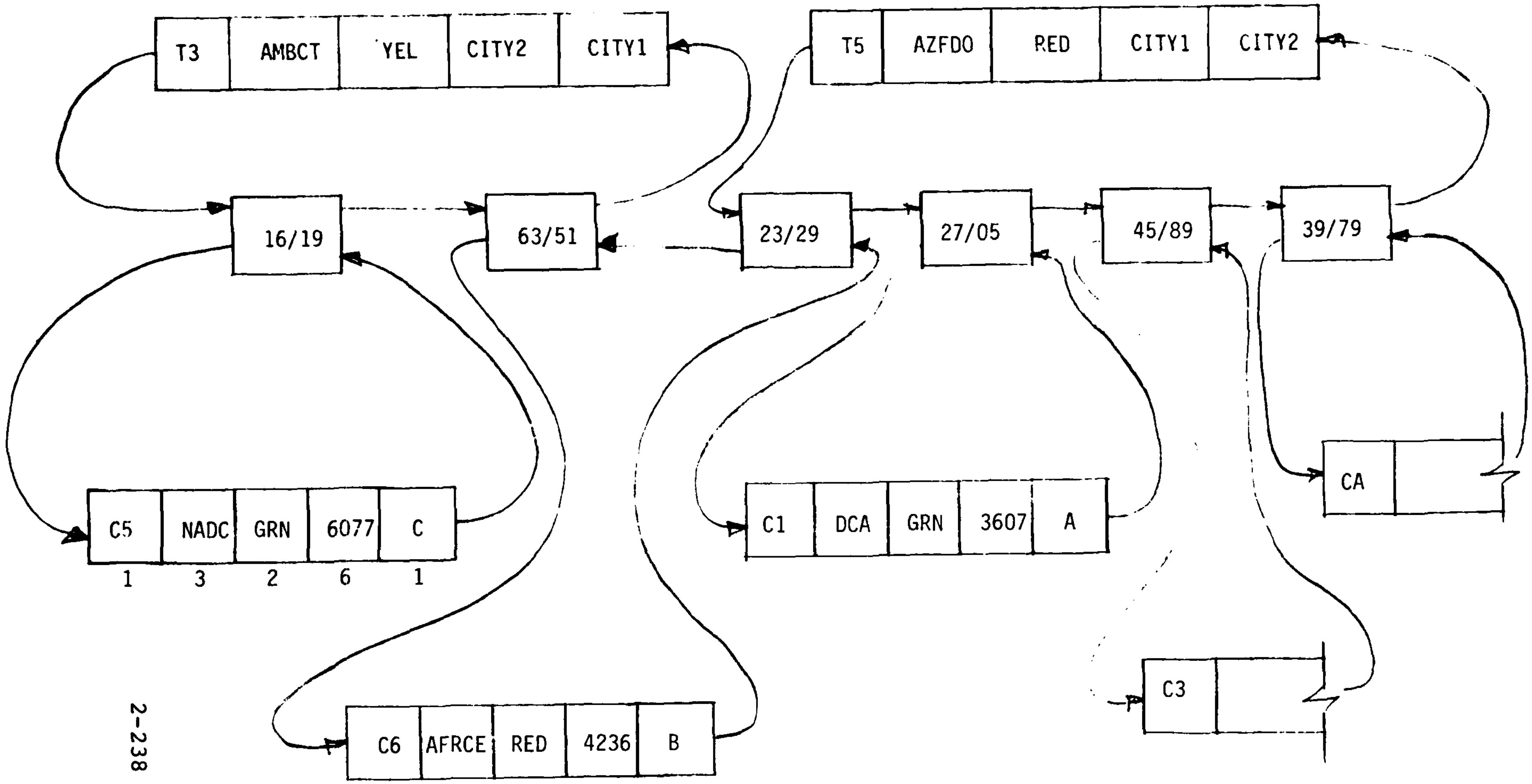


Figure 2-32. Network Model of Circuit/Trunk Sample Data

Q1 - Find the numbers for the circuits which use trunk T3
Q2 - Find the number of the trunks which host circuit C6
R1 - Step 1 - Find supplier with T#=T3
Step 2 - Find "next" link for this trunk
Step 3 - Link found? (if not, exit)
Step 4 - Find circuit for this link
Step 5 - Get circuit record
Step 6 - Print C#
Step 7 - Go to step 2
R2 - Step 1 - Find circuit with C#=C6
Step 2 - Find "next" link for this circuit
Step 3 - Link found? (if not, exit)
Step 4 - Find trunk for this link
Step 5 - Get trunk record
Step 6 - Print T#
Step 7 - Go to step 2

We can see that with the network approach, symmetric questions require symmetric answers. This is an advantage over the basic hierarchical approach. However, there may be a strategy problem if we are asked to find the patch link record between trunk T3 and circuit C6. Do we start from the trunk and traverse its chain or from the circuit and traverse its chain.

The network model overcomes most of the difficulties encountered with the storage operations in the hierarchical model.

(1) Adding a new record is easy.

(2) Deletion of a record can be accomplished without the loss of additional data.

(3) Updating is simplified since redundant information is minimized or eliminated.

The problems of the hierarchy disappear mostly because of the particular form the network takes. The problem is really one of data normalization. The major disadvantage of the network model is that it is too close to the storage structure. The user has to be thoroughly aware of which chains do and do not exist, and DML programming can become extremely complex. More significantly, the chains are directly visible to the user and must be directly represented in storage somehow. There is a risk that the user will become locked into a particular storage structure, contrary to the aim of data independence.

2.10.4.3 The Relational Approach--The relational approach is based on the mathematical theory of relations. This provides a sound theoretical foundation. On the other hand, the terminology employed is taken directly from the theory, so that in some places the user is faced with having to learn new terms for familiar concepts.

Table T (Figure 2-30) actually represents a relation which we will call a "trunk" relation. Each table row is called an "n-tuple" where n is the number of columns in the row. The columns in the relation identify the "domains" over which the relation is defined. The trunk relation is defined over the domains T#, NAME, STATUS, SOURCE, and TERM. Note that:

(1) No two tuples (rows) are identical.

(2) The ordering of the tuples is insignificant.

(3) The ordering of the domains (columns) is insignificant.

(4) Every value within a relation is a unit data item (the principal of normalization).

From the user's point of view the relational model of a data base is a collection of normalized relations of assorted degrees. (The degree of a relation is the number of domains over which it is defined.) The DML for a relational data model is somewhat different. The relational DML is based on the fact that queries against the relations that make up the DB result in another relation defined by a "relational algebra". A relational algebra operation is one which takes one or more relations as its operands and produces a relation as its result. There are a number of such operations, two are illustrated.

To "project" a relation over specified domains the unprojected domains are eliminated and then the redundant rows are eliminated. This is illustrated in Figure 2-33 which is the projection of the trunk relation over the domains of STATUS and TERM. Note that the tuple that resulted from T#=T4 was eliminated since it would have been redundant with that which resulted from T#=T1.

Two relations with a common domain, D, can be "joined" over that domain. The result is a relation in which each tuple consists of a tuple from the first relation concatenated with a tuple from the second relation which contains the same D-value. Figure 2-34 shows how Table C and Table TC (from Figure 2-30) are joined over the domain C#.

Let us now consider several queries on the relational data base.

STATUS	TERM
GRN	CITY4
GRN	CITY3
YEL	CITY1
RED	CITY5

Figure 2-33. Projection of Relation Trunk Over (STATUS, TERM)

USER	STATUS	PHONE#	MAINT	C#	T#	PATCH
DCA	GRN	3607	A	C1	T1	36/08
DCA	GRN	3607	A	C1	T5	27/05
NADC	GRN	1245	B	C2	T1	18/97
NAVY	GRN	7093	A	C3	T2	72/81
NAVY	GRN	7093	A	C3	T4	19/27
NAVY	GRN	7093	A	C3	T5	45/89
ARMY	YEL	8512	B	C4	T4	56/40
ARMY	YEL	8512	B	C4	T5	39/79
NADC	GRN	6077	C	C5	T3	16/19
NADC	GRN	6077	C	C5	T1	05/63
AFRCE	RED	4236	B	C6	T3	63/51
AFRCE	RED	4236	B	C6	T1	95/80
AFRCE	RED	4236	B	C6	T5	23/29

Figure 2-34. Two of Relations TC and C Over Domain C#

Q1 - Find the circuits which are serviced by trunk T4.
Q2 - Find the trunks used to support circuit C6.
Q3 - Find the trunks which provide a path from CITY2 to CITY1.
R1 - Step 1 - Join TC and Z1 over T#.
Step 2 - Project the result over C#.
These steps are illustrated in Figure 2-35.
R2 - Step 1 - Join TC and the unary relation Z3 over C# where Z3 is defined by:

Z3
C#
C6

Step 2 - Project the result over T#
R3 - Step 1 - Join T and Z4 over SOURCE
Step 2 - Join the result and Z5 over TERM
Step 3 - Project the result over T#
Z4 and Z5 are defined by:

Z4	Z5
SOURCE	TERM
CITY2	CITY1

These examples should serve to illustrate the relational algebra approach. The user actually constructs a set by performing a sequence of set operations. Relational algebra is a procedural DML approach.

The advantages of the relational DB should be clear.

(1) Adding new relation or tuples to existing relations is trivial.

(2) Deletion of a relation or a tuple is easily accomplished.

(3) Updating of information (a data item) affects only a single location in the DB.

(4) Total data independence can be achieved.

The overriding disadvantage of the relational approach to DB is that no large scale implementations have been proven effective. However there is evidence that some are soon to be available. In addition the effective use of a relational DB requires some insight to set up the right normalized relations.

2.10.5 The System Control Data Base Classified

Now that some of the background data and characteristics of various data base concepts have been summarized, we can begin to analyze the system control application to see what aspects the

DBMS must present to adequately support the information processing requirements. In this respect, choosing the right DB

Z1

T#
T4

a) Unary Relation

Z2

C#	T#	PATCH
C3	T4	19/27
C4	T4	56/40

b) TC Joined with Z1

C#
C3
C4

c) Z2 Projected over C#

Figure 2-35. Response to Querry Q1.

approach for system control is largely a matter of determining all the requirements of the application. In general that is not realistically possible at this point in the development. As an alternative we can address the requirement of primary importance at this point in time. The alternate route algorithm. Since this will be the primary or highest priority DB information requirement, DB design from this perspective will certainly optimize the DB with respect to that dimension.

The connectivity data base shown in reference 2 shows a DB design that is somewhat optimized from the perspective of the network control technician display requirements. Ultimately, a DB that supports both these dimensions and any others that might arise later would be highly desirable. Keeping in mind the arguments made earlier about DB classification (Paragraph 2.10.4) a relational DB would be the best choice. However as pointed out, no large scale implementations of a relational data base are yet available. They are still "state-of-the-art". This is due, not so much because of the theoretical aspects of the DML but because of the complexity of implementation of the relational calculus required by the DML. Unfortunately practical relational data bases still exist only as concepts and theories.

There are numerous examples of both network and hierarchical data bases. Both structures are capable of supporting the system control application. However, it is quite easy to show that a network DB requires less application software to be developed just for the altroute algorithm than a hierarchy. The biggest reason for this is that a good deal of information (mainly inter-record dependency) is stored in the set linkages of a network DBMS. The hierarchy does not directly support this capability and it must be maintained by application software. This argument holds only in the case that the application DB requires these set linkages for operation.

It should be fairly obvious from previous discussion of the alternate route solution algorithm that multiple set relationships are required. It is a fact that the files within the system control DB contain multiple many-to-many relationships which are most directly supported by a network DB. Implementation with a heirarchical DB requires multiple inversion lists. The management of inversion lists requires more processing horsepower than the relinking of pointers, especially when frequent updates are made. At this point, one must be careful to observe that updates to the system control DB will occur for two reasons. The most obvious is for the addition of new circuit information to the DB. This should occur relatively infrequently. However another class of update occurs when alternate routes are implemented or primary routes restored, etc. These updates will occur relatively frequently with respect to the first kind of update. Therefore, although the static connectivity DB may change relatively infrequently, dynamic operation of the network has the potential of requiring frequent

but temporary modifications to the relationships between records in the DB. The alternate route algorithm is evidence of this. The system control data base must support frequent updates as well as rapid retrieval of information. Networks are much better suited to achievement of these two goals simultaneously than hierarchies are.

2.10.6 DB Structure Design

This subsection provides a DB structure design based on the information requirements of the alternate route algorithm. This algorithm, as described earlier, spends a considerable amount of time and processor power on the task of manipulating pointers and remembering the state of records in the DB so they can be restored later. This algorithm is based on the connectivity DB as shown in reference 2. The use of the DB design presented in this section will result in a significant reduction in the algorithm coding required. This is because the proper use of a DBMS system will provide all pointer manipulation as a standard feature.

There are two key goals to be strived for in the DB structure design to optimize the alternate route selection algorithm. These have to do with organization of data into records and definition of the inter-record relationships.

(1) The record content and inter-record relationships should be defined so that as much information as possible is carried in the record relationships rather than the records. This maximizes the amount of data management done by the DBMS and minimizes that which must be done by the algorithm software.

(2) The data within the records should be partitioned so that all data items within a record are related to the extent that they "always appear together" and never "always appear with a data item from another record". As a secondary effect of this goal the data items should be defined such that changes in the structure of network represented can implemented by changes in the data item content of one or more records rather than by modification of the linkages that maintain the relationships or sets. Achievement of this goal will result in the fastest possible speed of responding to real time changes in network being controlled.

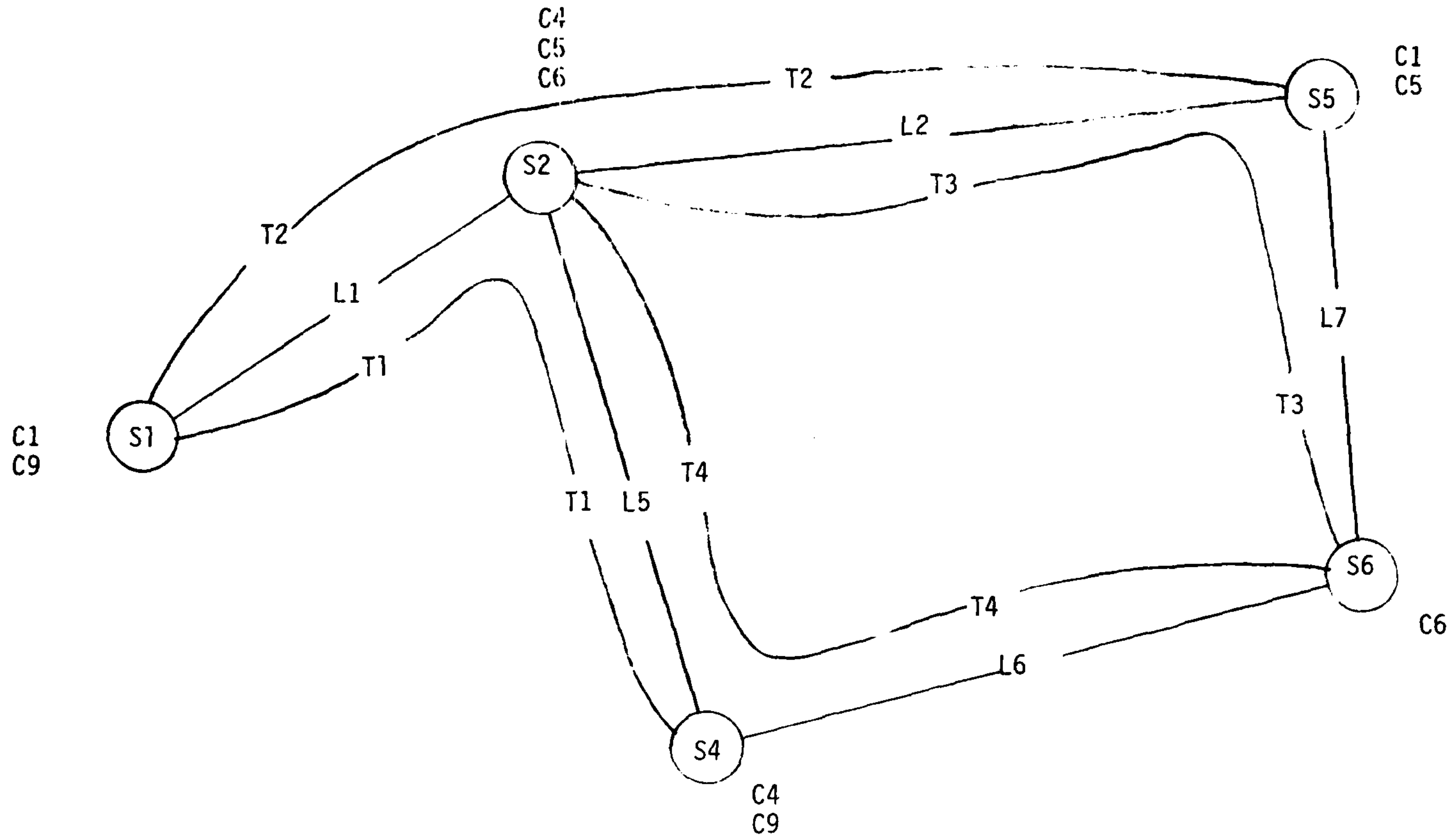
Achievement of these two goals simultaneously is generally not possible since they are somewhat in conflict. The best we can do is hope to achieve some balance in their applications.

Let's look for a minute at the content of the DB and the content of the data items outlined for the connectivity DB of reference 2. It can be seen that most of the records shown there contain,

as data items, the names or identifiers of other records in the DB. If these data items can be converted to inter-record relationships then we have come a long way to achieving our goals. As mentioned earlier, the European DCS structure provides multiple many-to-many relationships. The DB structure must model this. At the same time the central transform of the altroute algorithm depends on the ability of the DB to provide information about all possible routes between two stations. Therefore, the central inter-record relationship in the DB must be between stations and links. This is a many-to-many relationship since a link serves multiple stations and a station is served by multiple links. By way of example, Figure 2-36 shows several stations connected by links. In addition, there are trunks that connect stations using various links on the way. There is a many-to-many relationship between trunks on links. In addition, there is a many-to-many relationship between the trunks and the circuits which ride them. In addition these relationships are all related to each other. All of these DB entries - the circuits, trunks, links, etc. - have the common attribute that they use communication bandwidth over a link. To model this with the system control DB, the concept of the "profile element" is presented. A profile element (PE) can be a record in the DB which is the primary inter-record relationship between circuits, trunks, and links. Under this concept the circuit connectivity path becomes a chain made up of PE's. The DB PE file will have owners from the circuit, trunk and link files. So, using the shorthand notation introduced earlier, the central portion of the DB will look like that shown in Figure 2-37. The information contained in these records will be basically as shown in Tables 2-9, 2-10, and 2-11 respectively. The PE records will be discussed shortly. The record content shown in these Tables is substantially less than that shown in Tables 3-15 through 3-17 of reference 2. The information that has been deleted from data items is now present in the chain membership of new DB structure.

To see how this reduction in record data item size has been accomplished, let us examine a little more carefully the structure shown in Figure 2-37 in the context of the sample network structure shown in Figure 2-36. Remember that the circuit file in Figure 2-37 actually represents multiple occurrances of circuit records each of which own a chain of profile elements. Figure 2-38 shows an example of the profile elements that would be owned by the circuit record that represents the circuit designated C9 from station S1 to S4 in Figure 2-34. Notice that each PE record also belongs to a chain owned by a trunk and a link record. Of course, each record in the DB contains some kind of data which maintains the chaining sequence, but the DML manages that data and it is, in fact, masked from the DB user. The DML for network DBMS's make it easy to identify the owners of all chains any specific record belongs to. Therefore, for the structure shown, it is an easy task to identify all the trunks used by a circuit between point A and point B. It is equally easy to identify all the circuits served

by any trunk or link. Using the DML to traverse the circuit owned chain (Figure 2-38) we can easily find out that circuit C9 rides on trunk T1 while on link L1 and also rides on trunk T1 as it rides on link L2.



T1 - Trunk Designator
L5 - Link Designator
S3 - Station Designator
C7 - Circuit Endpoint

Figure 2-36. Sample Network Model

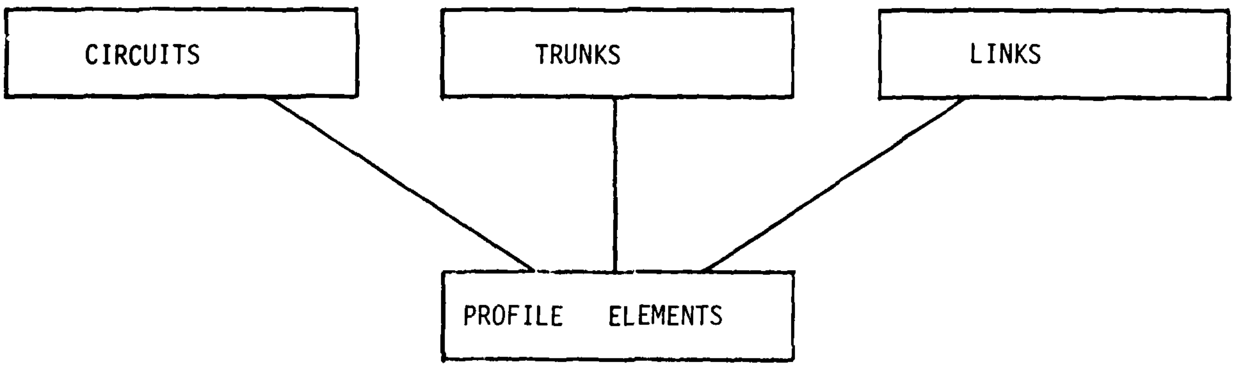


Figure 2-37. Central System Control DB

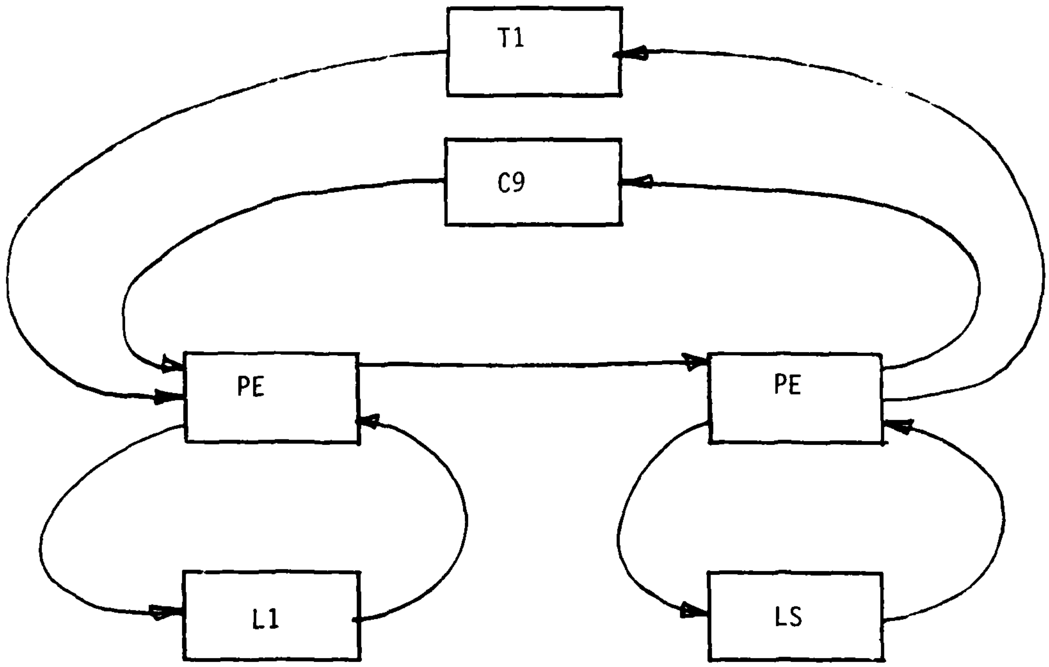


Figure 2-38.Circuit C9 Sample Profile

Figure 2-39 shows the entire circuit, link, trunk, profile element data base that represents the network in Figure 2-34. Even for this simple network, the structure is quite complex. Therefore, as the other files in the DB are added, the shorthand notation (like that in Figure 2-37) will become quite useful. Note that the chains shown in Figure 2-39 represent only the forward (next) chain pointers as shown in Figure 2-25. Since this is the central portion of the DB, it is quite likely that these sets will have prior and owner pointers as well which will make Figure 2-39 even more complex.

Until now, the profile element records contain no user data items, only chain pointers. Additionally, the PE's used so far represents only those circuit, link, trunk correlations that are actually current and active. The extensions based on the PE concept give the DB the ability to represent preplanned altroutes using the same PE structure. A chain belonging to a certain circuit can contain PE's for inactive segments as well as active ones, and the PE record itself can contain a data item which states whether it is active or inactive. Therefore, in some cases the DB content may be changed to reflect the implementation of an altroute by changing the content of data items in the PE records for the circuit affected. This is consistent with the two goals stated at the beginning of DB design. The set linkage contains all the connectivity information, but the current state of the systems was changed by modification of data items. The DML was not required to juggle pointers to logically move PE records. It is likely that the PE record will contain other data items for various purposes. A best guess at the data item content for this record type is shown in Table 2-12. These arguments do not preclude the idea that PE's can be logically moved from one circuit to another, however proper implementation of the algorithm can minimize this.

The central idea of the system control DB is now complete. Refinement of the linkages involved and data item content is, of course, required. However, the potential savings in both DB size and algorithm speed are evident. These two issues are addressed briefly later, but the other files in the DB and their content also needs to be identified.

All of the information in the connectivity path file of the old data base is now stored in the linkages of the new data base. The connectivity path ID is now the Profile Element ID, and the terminating stations are the endpoints of the chains. This file, therefore, no longer exists. The relationship of the, Sector file, Node file, and Station files are somewhat hierarchial in nature and are represented that way in Figure 2-40 which shows the entire DB structure in shorthand form. The Faults file is shown linked to virtually every other record in the DB. Logically, it needs to be linked to only the circuit, trunk, or link file, because these imply which station, nodes, and sectors are affected. However, the linkages support the reports on

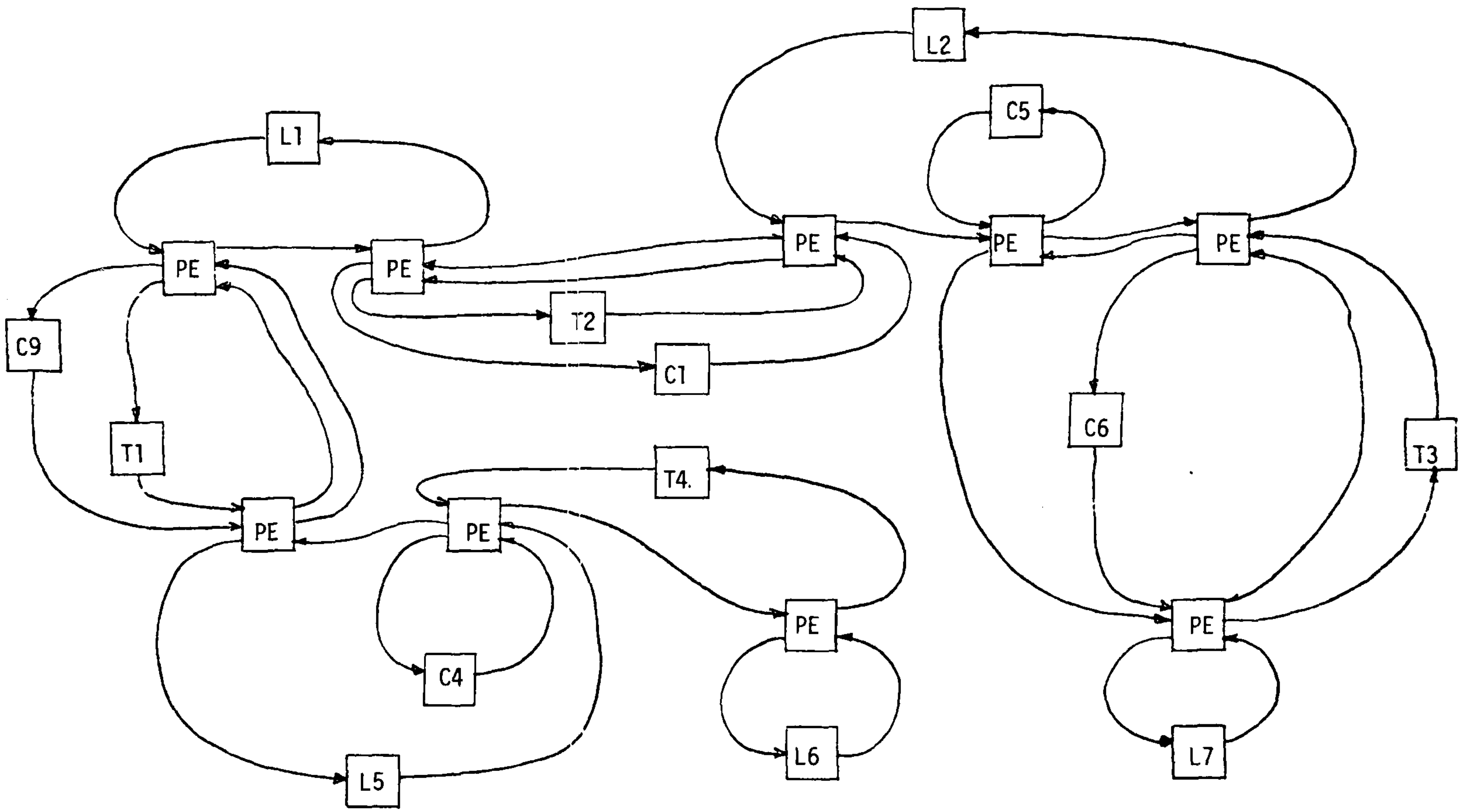


Figure 2-39. Sample Network DB Chain Structure

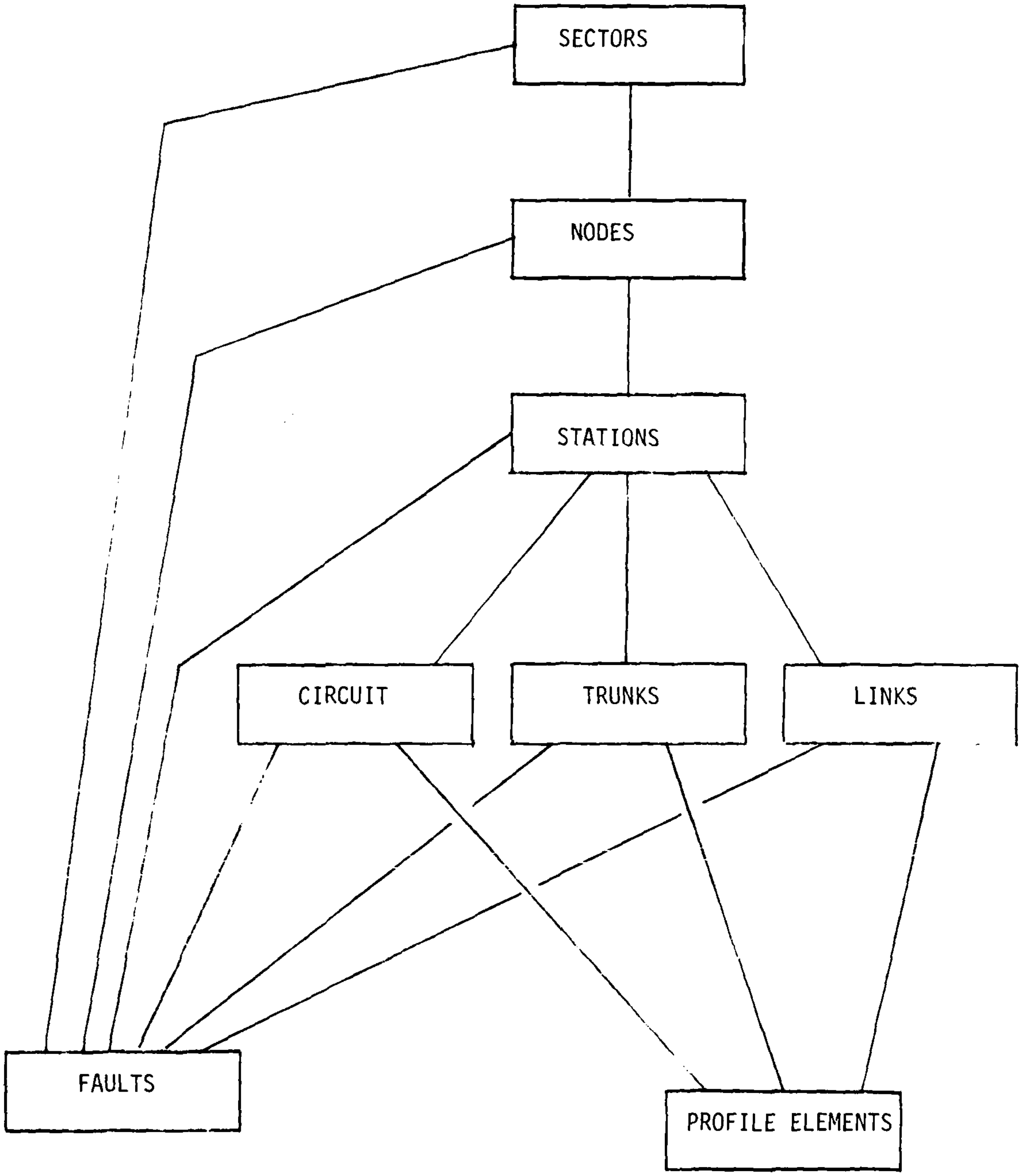


Figure 2-40. System Control Netowrk DB Structure

current fault activity which will almost certainly be part of the system control operations. The Fault file content is shown in Table 2-13. For this file, the ordering of faults with respect to stations, nodes and sectors can now be maintained by their order within a set.

The content of the sector, node and station files is as shown in Tables 2-14 through 2-16.

2.10.7 Data Base Size

With the work done so far, it is easy to estimate the DB size. The true size of each record type is determined by the size of the data item fields (from Tables 2-9 through 2-16) and the number of pointers the DBM must maintain. The size of each pointer field will be two bytes and the number of pointers depends on the number of chains the record belongs to and whether the chains have prior or member pointers. (Next pointers are always required.) Additionally, a pointer field will be required if a record is to be retrievable as a random entity from the DB. If worst case is assumed, the number of bytes required for pointers for each record will be (3N+1 pointers) x (2 bytes/pointer), where N is the number of chains the record participates in. Table 2-17 summarizes the record size content for the DB. Record count estimates are based on the old DB and as a result, the new DB size is reduced 36% from the old - A significant savings.

2.10.8 Summary of DB Considerations

We have attempted to show how application of modern DB techniques to the system control problem can result in improvement of DB efficiency in terms of both DB size and algorithm performance. The result of the preceeding sub-paragraph has demonstrated the size savings. Algorithm throughput savings are more difficult to estimate. However, a two or three to one improvement could be expected based on algorithm simplification resulting from an optimally designed DB. Throughput performance depends, to a great extent on operation of the DBMS software. Further refinement of the techniques applied here are required to totally optimize the system control DB.

TABLE 2-9. LINK FILE CONTENT

Item	Comments	Bytes
Link ID		5
DOD (Direction 1)	Degree of degredation (i.e., out or degraded)	1
DOD (Direction 2)	Same as for Direction 1.	1
ETR and DTG	Estimated Time to Restore and Date/Time group when Fstimate was made.	11
Highest RP	Highest restoration priority carried by the link to establish criticality of temporary permanent restoral.	2
HAZCON		1
Data Base Distribution	List of all stations (2), nodes (2), sectors (2), and areas (2) to receive DB updates for this link. Use addressing as in ATEC 10000 Spec.	24
	RECORD SIZE (BYTES)	45

TABLE 2-10. TRUNK FILE CONTENT

Item	Comments	Bytes
Trunk ID		6
VFCT CCSD	Cross reference to VFCT identifier if this is a VFCT.	8
Reroute ID #1 and Flag	Identifies the trunk which is preplanned for restoral of this trunk, and whether it is activated.	7
Reroute ID #2 and Flag	Identifies either a trunk other than the preplanned reroute which was used to restore this trunk, or a trunk which has preempted this trunk. A flag indicates that this field is idle, or that this trunk has been rerouted or preempted.	7
Degree of Degradation (DOD), Direction 1 and Fault Location	Identifies whether entire group or partial group in direction 1 is affected, whether this is a partial degradation, out of service or a HAZCON.	4
Degree of Degradation (DOD), Direction 2 and Fault Location	Same as above, except it is for direction 2.	4
Data Base Distribution	List of all stations (6 x 3), nodes (3 x 4), sectors (3 x 4), and areas (2 x 3) to receive DB updates. Use addressing as in ATEC 10000 Spec.	48
Control Responsibility		3
Networks Impacted (VON, DIN,...)	Identifies which control functions need the data.	2
PMP		
Related Route		6
Monitoring Rgrd Flag		1
	RECORD SIZE (BYTES)	96

INTENTIONALLY LEFT BLANK

TABLE 2-11. CIRCUIT FILE CONTENT

Item	Comments	Bytes
User	Identifies name of person to contact relative to this circuit.	12
Phone Number	Permits calling user.	10
RP	Restoration Priority used in impact analysis of outage.	2
Preempting CCSD and Flag	Identifies that this circuit has been preempted by the identified circuit.	9
Reroute ID #1 and Flag	Identifies the circuit which is preplanned for restoral of this circuit, and whether it is activated.	9
Reroute ID #2 and Flag	Identifies either a circuit other than the preplanned reroute which was used to restore this circuit which has preempted this circuit. A flag indicates that this field is idLe, or or that this circuit has been rerouted or preempted.	9
Degree of Degradation, Direction 1, and Fault Location	Identifies whether there is a complete outage or a degradation, and where the fault is. Direction 1 for circuit level faults.	4
Degree of Degradation, Direction 2, and Fault Location	Identifies whether there is a complete outage or a degradation, and where the fault is. Direction 2 for circuit level faults.	4
Control Responsibility		3
	RECORD SIZE (BYTES)	62

TABLE 2-12 PROFILE ELEMENT FILE CONTENT

Item	Comments	Bytes
Profile Element ID		2
Active/Inactive	Indicates whether PE represents a currently active segment.	1
Permanent/ Temporary	Is PE segment a permanent part of profile for circuit.	1
Altroute Level	If PE is representing an Altroute what level of preemption is it.	1
Miscellaneous Info		3
	RECORD SIZE (BYTES)	8

TABLE 2-13. FAULT FILE CONTENT

Item	Comments	Bytes
Fault ID		6
DTG (of original report)		7
Severity	Link, trunk or circuit level.	1
Direction	Direction of outage.	1
RFO	List of each reported, up to 3.	9
ETR and DTG	The estimated time to repair and the time at which the report was made.	11
DOD	Degree of degradation.	1
DTG of Fault Closure		9
Station Submitting Closing Report		3
RP of Highest RP	Serviced by failed capability.	2
Comments	Narrative field of fault report.	80
	RECORD SIZE (BYTES)	130

TABLE 2-14. SECTOR FILE CONTENTS

Item	Comments	Bytes
Sector ID		3
ACOC ID	Locates the sector within the global data base.	3
Sector Reporting Status	Indicates if any reports are overdue or if the telemetry from the station is out of service.	1
CCSD of ATEC Telemetry to ACOC	Permits the details of that telemetry path to be looked up if it must be restored or its condition is in question.	8
	RECORD SIZE (BYTES)	15

TABLE 2-15. NODE FILE CONTENTS

Item	Comments	Bytes
Node ID		3
Responsible Area	Locates the node within the global data base.	3
Node Reporting Status	Indicates if any reports are overdue or if the node-sector telemetry is out of service.	1
CCSD of ATEC Telemetry to Sector	Permits the details of that telemetry path to be looked up if it must be restored or its condition is in question.	8
	RECORD SIZE (BYTES)	15

TABLE 2-16. STATION FILE CONTENTS

Item	Comments	Bytes
Station Name		3
Station Status	Indicates if the station is totally or partly out of service or if a HAZCON exists.	1
Responsible Area	Locates the station within the global data base.	3
AUTODIN Site Flag	Indicates that an AUTODIN switch is at this site, used in status display generation.	1
AUTOVON Site Flag	Indicates that an AUTOVON switch is at this site, used in status display generation.	1
ATEC Equipped Flag	Indicates that ATEC exists at this site, used to determine if communications with ATEC are possible.	1
Manned/Unmanned Flag	Indicates if the station is a manned site, to determine what actions are possible or if there can be communications with an operator.	1
CCSD of ATEC Telemetry to Node	Permits reviewing that circuit to determine how it can be restored or other items relative to its operational status.	8
Station Reporting Status	Indicates that the telemetry to the site is out of service or that reports are overdue.	1
Time REport Due	Indicates the the time that an overdue report should have been submitted.	4
	RECORD SIZE (BYTES)	24

TABLE 2-17. DB SIZE SUMMARY

RECORD NAME	DATA ITEM BYTES	NO. CHAINS	POINTER BYTES	BYTES PER RECORD	RECORD COUNT	BYTES PER FILE
Link	45	3	20	65	410	26,650
Trunk	96	3	20	116	1,250	145,000
Circuit	62	3	20	82	10,500	861,000
Profile Element	8	3	20	28	25,000	700,000
Fault	130	6	38	168	3,600	604,800
Sector	15	2	14	29	5	145
Node	15	3	20	35	30	1,050
Station	24	5	32	56	100	5,600
TOTAL DB SIZE (BYTES)						2,344,245

SECTION III

SWITCHED NETWORK CONTROL

3.0 INTRODUCTION

This section discusses switched network control, in particular control of the AUTOVON network. Control is applied to a circuit switched network either to enhance critical subscriber connectivity, improve the grade of service (GOS), or the insure the stable operation of the network. Because the network is extensively engineered based on historical traffic data, and because it provides multiple routes between any two users, no great improvement in GOS can be expected from applying control. Therefore, this possible reason for controls is not discussed in this report. The remaining two reasons and controls resulting from them are discussed in the following paragraphs.

3.1 GENERAL CONSIDERATIONS

3.1.1 Traffic Control

Traffic control is the process of artificially restricting the network traffic in some manner, ostensibly to either protect the network or to optimize overall network operations. Examples of traffic controls are the following:

- Destination code cancellation, where all calls to a specific switch are blocked
- Line load control, where a subset of subscribers are denied dial tone
- Trunk directionalization, where a set of trunks is converted from two way to one way operation, thereby restricting the pattern of traffic flow.

These controls were developed over the years because it was observed that if uncontrolled overloads were allowed to enter the network, the amount of traffic actually carried sometimes decreased dramatically. The controls are various means which restrict the network traffic, thereby preventing overloads of the magnitude required for the precipitous drop in carried traffic, but they were subjectively designed and applied without a firm understanding of the cause and nature of the reduction of call carrying capacity in an overloaded network.

In an ideal network of trunks, this reduction in call carrying capacity would not occur. Rather, as the offered traffic increases, the carried traffic would also increase regardless of traffic level. The carried traffic would not increase as fast as the offered traffic, resulting in increased blocking, but the carried traffic would never decrease. In this situation, there is never a need to artificially restrict the traffic.

Experience with real networks has shown that if the traffic increases above some level, the portion of network capacity being used for signalling which results in the call being blocked, called the ineffective signalling, increases in a non-linear manner. This results in more traffic being blocked because there is less network capacity for carrying calls, which in turn causes more ineffective signalling. This phenomenon has been analyzed for portions of the Bell system (reference 14) which use the 4A-ETS switching system. In this case, it was shown that the call carrying capacity of the network could be reduced to very low values by traffic overload. Furthermore, it was discovered that for some levels of offered traffic, two quasi stable states could exist. One state is characterized by low call blockage and a small amount of ineffective signalling, while the alternative state has excessive blocking and a large amount of ineffective signalling. The system appeared to transition from the normal state to the thrashing state by the introduction of a severe traffic overload, but a transition back to the normal state did not occur as a result of reducing the traffic. Rather, a spontaneous transition occurred some time later.

The AUTOVON system has different routing rules and the switches operate differently, but the same phenomenon is possible. In the AUTOVON system, the originating office equipment is held only long enough to connect across the tandem office, and only the primary route is searched at the tandem, as opposed to the Bell 4A-ETS system where originating office equipment is held only until a tandem office is reached, and all routes from the tandem office are searched. These procedures are very different, but overall AUTOVON holds office equipment at least as long as the Bell 4A-ETS network.

A factor in AUTOVON which aggravates the thrashing situation is the handling of trunk glare, i.e. the situation where the same trunk is seized from both ends simultaneously. The 490L switch drops both ends and goes back into the trunk searching routine. In an overloaded network, particularly one which has some damaged transmission facilities such that bottlenecks exist, this method of glare response can lead to entire groups of trunks filling up with glare conditions. With many calls attempting to seize the trunk group from both ends, the calls cycle through the glare/release sequence several times before succes sfully seizing a trunk. This increases the time that each call holds switch equipment, aggravating the thrashing tendency.

The special nature of the AUTOVON system's mission makes it particularly important to avoid a thrashing situation. The AUTOVON system has a multilevel priority system which allows critical users to preempt trunks being used by less important traffic. This preemption is performed by the seizing switch (either originating or tandem), which then uses the trunk to signal to the next switch. Until a receiver is seized in the next switch, it has no knowledge of the calls precedence. Therefore, a precedence call gets no special treatment relative to receiver seizure, i.e. receiver seizure is a precedence blind operation. In a thrashing situation, one of the major causes of call blockage is a trunk timeout waiting for a receiver. Therefore, in a thrashing network high priority calls can easily be blocked. This is completely unacceptable, and must be prevented.

Another precedence blind operation which needs to be controlled somehow is the initial seizing of a digit receiver by the subscriber's loop. While this problem cannot cause call blockage, delays which are unacceptable for high priority users can occur during periods of traffic overload. Some control is needed to prevent these delays.

Except for these delays and defects in call setup procedures, a circuit switched network is completely self regulating. Traffic controls are needed only the prevent the non-ideal nature of the call setup procedure from taking over the network. This take over or thrashing phenomenon is not well understood, but it has been shown to be dependent on two factors - the length of time taken by signalling and the timeout of trunks due to receiver non-availability. Both of these factors can be eliminated by the use of high capacity common channel signalling. At some point, the ratio of signalling baud rate to the number of trunks becomes great enough to insure that trashing does not occur. This is also true for in band signalling. Certainly, when every switch has enough receivers so that one can be assigned to each trunk, timeouts waiting for a receiver cannot occur.

3.1.2 Network Control

Network control is primarily concerned with the allocation of transmission resources among the competing users. As such, it is primarily a transmission system function. However, the common user switched networks have some peculiar needs and capabilities which if utilized appropriately can improve the overall service to users of the DCS. Furthermore, an aspect of network control, routing control, is unique to the common user network.

Resource allocation is currently performed completely independent of the common user network. If a piece of the transmission system fails, an attempt is made to reroute the circuits it carried according to an almost static priority system. AUTOVON trunk groups typically contain about half high priority circuits

AD-A080 767 HONEYWELL SYSTEMS AND RESEARCH CENTER MINNEAPOLIS MN F/G 17/2
SYSTEM CONTROL FOR THE TRANSITIONAL DCS.(U)
MAY 79 F C ANNAND, D E DOTY, R V KRZYZANOWSKI DCA100-78-C-0017
UNCLASSIFIED TR-3 SBIE-AD-E100 328 NL

4 OF 4
AD A080767

END
DATE FILMED
3-80
DDC

and half low priority, so that some connectivity is always assured. However, the importance of any particular piece of connectivity in AUTOVON is highly dependent upon the status of the rest of the connectivity - deletion of a single intra European trunk group has very little impact on the network performance. Furthermore, preempting part of one interswitch trunk group to restore part of another, which could occur under the current procedure, would result in service to the user which is worse than that obtained by leaving the system alone. This is an extreme case, but it serves to demonstrate the advantage of considering overall AUTOVON connectivity in the network control function.

In order to utilize the capability of the network to operate effectively in the face of transmission resource failures, or switch equipment failures, it may be necessary to modify the routing used by the network. AUTOVON routing tables provide only two or three possible routes. Multiple transmission system failures could disrupt all of these routes, yet leave enough of the network intact to have a source destination path.

The existence of a routing control can also make certain other controls practical. Specifically, in a network with heavy traffic loading, better overall service can be provided by restricting traffic to the primary route only. The primary route typically uses the minimum amount of network resources, and when there is enough demand to use all the network resources overall service is improved by minimizing the resources used by any particular call. This control is not practical in the current AUTOVON system because failure of a single group could isolate some source destination pairs. If routing control existed such that a real time response could be made to trunk group failures, restricting traffic to primary routing only would be an effective way to increase service to the user, especially on U.S.-Europe routes which are heavily used.

Another improvement in network operations which could be made practical if real time network control existed is the direct association of network logical connectivity with the physical connectivity. In the current network layout, certain transmission segments carry multiple trunk groups. Some of these trunk groups pass through a station containing a switch without terminating on that switch. There are several reasons for doing this - it reduces the switch processing load and the size of the switch matrix, it minimizes the number of switches (and hence degradation) in any given call, and most importantly it minimizes the impact of a switch failure on overall network operations. However, the splitting of the transmission resource into multiple trunk groups degrades the amount of traffic that can be carried at any given GOS.

If a network control existed which modified routing procedures in real time to use whatever connectivity exists, and which could

use the trunking resource of a failed switch to form new trunk groups between its adjacent switches, the primary need for having a logical connectivity different from physical connectivity would be eliminated. This would allow better service to be provided using less transmission resource, with no decrease in survivability.

Routing control can be achieved in several ways. Currently, a rudimentry form of routing control is achieved by reloading the entire switch memory from standby card decks or by use of the TDCS rapid memory reload function in case of specific failures. This method of control is relatively clumsy and requires a large amount of manual intervention. Furthermore, only a limited number of contingencies can be accounted for.

A much more flexible routing control would be possible if individual routing changes could be entered, especially if they could be remotely entered. This capability could be provided by the Rapid Access Maintenance Monitor (RAMM), by adding a communication line and adding a software module to make modifications to the RAMM disc files.

With a capability to monitor system status and to modify routing at the ACOC, its computer system could determine what routing changes have to be made to insure connectivity. Given the current switch equipment, only a centralized system has the visibility to make these determinations. This routing modification capability also provides the area controller much more flexibility in controlling the network without requiring action by local switch personnel.

The process of determining what routing changes need to be made in order to insure connectivity in spite of system failures could be distributed among the switches. The AUTODIN II network has this capability. The switches of AUTODIN II exchange information about the delay it takes to get to any destination. By collecting this data from each of the adjacent switches, a switch can determine the best route to take. IF a trunk is added its use is automatically determined by the next exchange of delay tables The operator does not even need to tell the system where the other end of the trunk is.

Adaptive routing analogous to AUTODIN II's capability could be provided for AUTOVON. This would require new computers for maintaining and performing the routing, and would require common channel signalling or some other interswitch data link for exchanging the routing information. As with AUTODIN II, there would still be a need for a network control center to monitor the system for things like psychotic switches, which claim that they can process all of the network traffic best, and other instances where manual intervention is required. Continuing the analogy with AUTODIN II, the network control center in this situation would not be needed to maintain either the operational or the

survivability characteristics of the network. If the routing control was exercised directly from the network control center as discussed previously, loss of the network control center would result in the loss of the capability to change routing procedures, and hence would be a reduction in network survivability.

Various adaptive routing schemes for circuit switched networks have been proposed (references 15 and 16). Most of these schemes were some sort of attempt to maximize network GOS over all traffic patterns. The reason adaptive routing is suggested here is to maintain network connectivity in spite of network failures. These goals may or may not be consistent. It is possible that a simple adaptation scheme will be more than adequate for the goals presented here.

Another special consideration in developing routing strategy for AUTOVON is the peculiar nature of AUTOVON traffic. Intra area traffic is relatively light, and a good GOS is easily maintained. Trans-Atlantic traffic is very heavy. to the point that alternate routing is often inappropriate.

To demonstrate this, consider a simple network model. In this network, there are three nodes. Two of the nodes originate an identical amount of traffic for the third node. They each have a direct trunk group of 10 trunks and , if blocked the traffic altroutes over an infinite group to the other originating node. This is a grossly simplified model of the trans-atlantic portion of AUTOVON, in that AUTOVON altroutes traffic to the three gateway switches. If we assume that the originating plus the overflow traffic at the node is approximately Poisson, we can compute a lower bound for the total amount of traffic blocked. This is an optimistic lower bound because in a real network there is blockage due to switching delays and a non-infinite trunk group between gateways in addition to excess blocking because of the deviation from Poisson.

The result of computing this bound is shown in figure 3-1. It can be seen that, for this case, alternate routing has less blocking only up to an offered traffic level of about 2 Erlangs per trunk. Above this level, primary routing only provides a better GOS, although not significantly so. However, this is an optimistic lower bound for the altrouting case. It also only considers the two trunks analogous to transatlantic trunks, and does not consider the affect on intra European traffic.

Since transatlantic trunks are designed for busy hour loads of greater than 4 Erlangs/trunk, this example demonstrates that during normal busy hour and especially during traffic overload periods, alternate routing should be eliminated

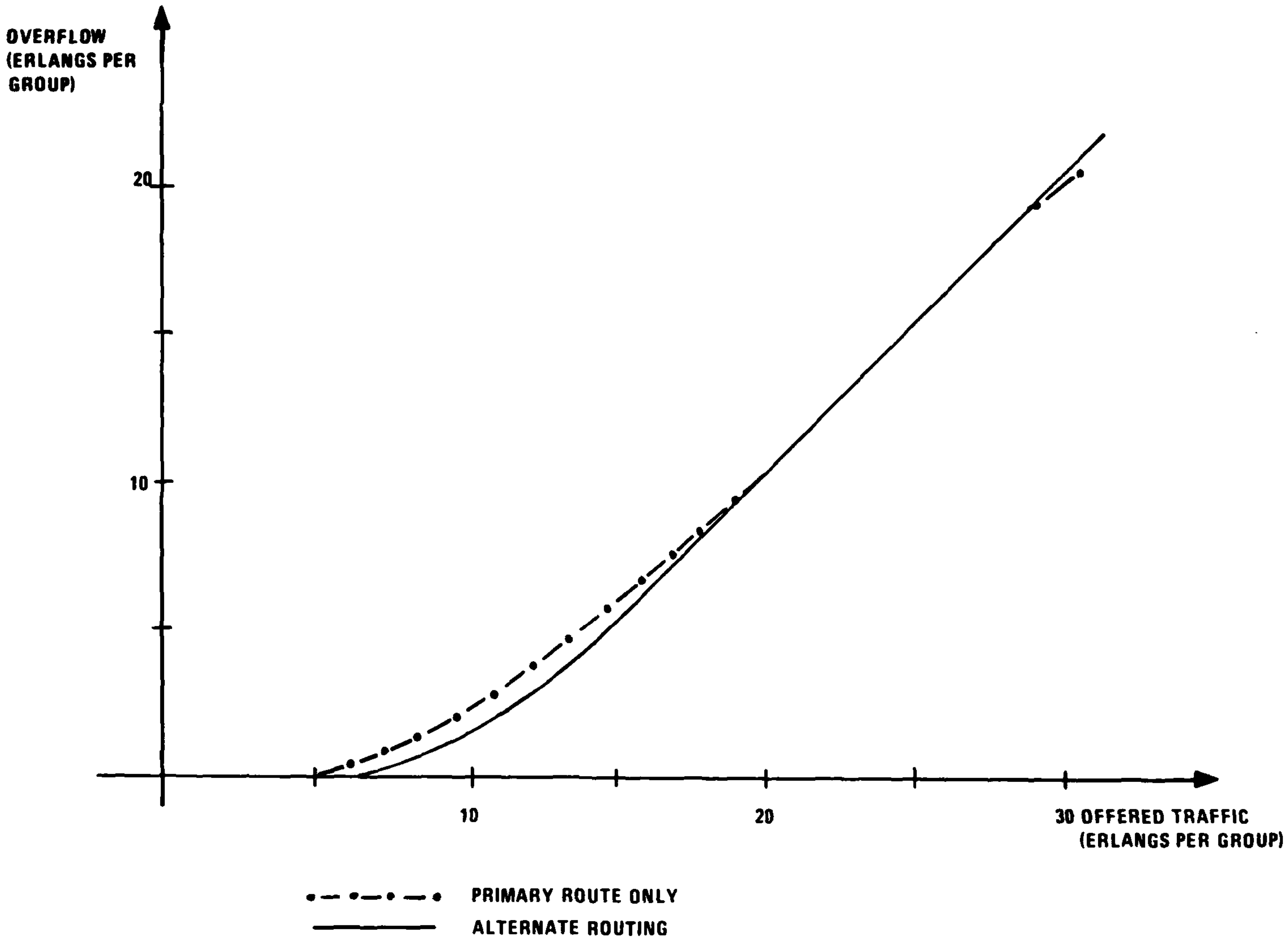


FIGURE 3-1. COMPARISON OF HIERARCHICAL AND PRIMARY ONLY ROUTING

for transatlantic calls. Furthermore, at any time when a portion of the network becomes overloaded, primary route only routing becomes more effective. The impact of using primary routing only on ineffective signalling traffic is perhaps even more significant. In the case of primary only routing, the MF transceiver is held for ineffective signals for either zero or 5 digits, depending on whether or not there is a direct trunk to the destination. If alternate routing is permitted, up to 16 additional digits may be sent by the originating switch before determining that the call is blocked. Freeing the network from the necessity for allocating resources (both trunking and switching resources) during the 16 digits will forestall the onset of the network thrashing condition.

Therefore, routing procedures are needed which switch to primary only routing any portion of the network which is overloaded, including transatlantic trunks during normal busy hour.

Since using primary only routing does not deny service and only slightly degrades GOS in normal traffic situations, the switch to primary only routing could be made based on less information than an service denying control.

For example, during each routing update, say every 5 minutes, the switch could determine to accept primary routed traffic on each trunk group based on its occupancy during the last update interval. This could be signalled back throughout the network, such that switches which would normally alternate route over these trunks would not try to do so. This technique of course requires a significant increase in the processing capability for route selection in the switch relative to the current system. If adaptive routing were installed, this control could be easily included in the adaptation design.

3.2 TRAFFIC CONTROL IN THE BELL SYSTEM

The Bell system is the largest, most complex telephone network in the world. It has developed in an evolutionary manner and now consists of many different types of switching equipment, from purely electromechanical analog switches to the FSS4, a modern stored program controlled digital switch. Because it is impractical to retrofit all of the Bell system switches with the most modern traffic handling equipment, traffic control in the Bell System must take into account the deficiencies of the older switches as well as the capabilities of the newer switches.

Significant automatic traffic controls were introduced to the Bell System in the late 1960's (References 17 and 18). At this time, simulation studies had identified the network thrashing problem and specific controls were introduced to prevent thrashing. Although other controls were used in some places to some extent, the primary controls developed were dynamic overload control (DOC) and trunk reservation equipment. DOC is somewhat

akin to ATOP in the AUTOVON system in that it senses the amount of common equipment busy and activates based on threshold crossing of that parameter. The major difference is that DOC affects tandem traffic whereas ATOP affects originating traffic. When DOC activates, it places a voltage on a special signalling line to all adjacent switches. The adjacent switches then refrain from attempting to tandem through the congested switch.

There are two types of trunk reservation equipments in the older Bell switches -- directional (DRE) and protective (PRE). These controls are activated when the number of trunks busy in a group cross a threshold. DRE is used only at the lower level end of a trunk group that goes between levels of the switching hierarchy. The best AUTOVON analogy would be that DRE is used by PBX's, since AUTOVON proper is a single-level switching hierarchy. When DRE is activated, the group is directionalized, i.e., the lower level switches do not try to sieze any of the trunks. PRE can be used on any trunk group from either end. When it activates, the switch only places primary routed traffic on the group. Alternate routed traffic is blocked from the group independent of whether there is a trunk or not.

In the Bell System, it is the prerogative of the individual operating company to use or not use any traffic control in their switches. Although these controls have been available for over a decade, they apparently have not sufficiently proved themselves (or perhaps their implementation is unreasonably complex) because they are not used very much (Reference 19). The capabilities of the ESS4 (Reference 20) have allowed improvements to be made in these controls, and allowed them to be included in the original design. Therefore, all ESS4 switches have automatic traffic controls.

The primary improvement in the control for the ESS4 is the inclusion of a more sophisticated data gathering process. For each area code and each switch code within three selected areas, the ESS4 keeps track of the probability of successful completion. It does this by counting the number of calls destined for each code, and the number which are eventually blocked by the network. If 70 out of 100 attempts are blocked, the code is declared hard to reach. If at some later time, less than 60 out of 100 are blocked, the code is removed from the hard to reach (HTR) list.

The ESS4 controls are equivalent to DOC and PRE, except that now three control states are possible rather than just two. When machine congestion (which now includes a measurement of unused processor real time as well as common equipment occupancy) exceeds a first threshold, adjacent switches refrain from routing HTR traffic to it. When a second, higher threshold is crossed, adjacent machines cease all tandem attempts to the congested switch just like the older DOC controlled. Similarly, the trunk reservation control, now called selective trunk reservation, has two thresholds. When the first threshold on number of trunks

busy is crossed, HTR traffic is no longer routed over the group. When a second threshold is crossed, all alternate routed traffic is restricted from the group.

Another ESS4 control is called automatic out of chain (AOOC) routing. This control allows a call to take an extremely circuitous route if it is blocked on its normal routes and there is very little network traffic in some part of the network. The classic example of this is the early Christmas morning traffic from New York to Florida which is routed via California, because all East Coast circuits are overloaded but the day hasn't yet started on the West Coast. These controls have been manually applied for years, but by using a travelling classmark on the CCIS network, low usage routes can be discovered and used automatically.

Bell system switches also have a full set of manual traffic controls, and extensive traffic data collection and display facilities to help network managers determine how to use them. A line load control function is provided to prevent lower level switches from gaining access to the switch. Some switches have line load control which, once applied, automatically cycles among lower level switches denying them access for five minutes at a time. Destination code cancellation is available for any 3, 6, 7 or 10 digit destination. Trunk groups can be manually directionalized for all traffic, or can be made directional for only HTR traffic. The automatic HTR list management can also be manually overridden, forcing certain codes to be HTR or exempting them from the HTR treatment.

3.3 OTHER COMMERCIAL NETWORKS

Other than the Bell System, not much information is available about network control procedures. Most international networks appear to control their networks by using manually-selected routing changes or service denials. An exception is the Japanese D10 system (Reference 21).

The D10 is a modern stored program controlled switch which has software registers for processing subscriber dialed data, and another set of registers for incoming tandem call data. When all of the registers are full, the scanners stop looking for service requests until a register is available. Switch overloads are detected by measuring the frequency of the scanner starting and stopping. If the overload threshold is crossed, all those calls awaiting service are locked out, i.e., the scanner queue is dumped. This provides the switch with the extra time needed to catch up on its processing, and the subscriber will probably reattempt within a few seconds anyway. On the trunk side of the switch, processing overload is usually a result of network thrashing anyway. Dumping the queue effectively reduces the time that network resources are used by a given call, thereby acting to stop the thrashing. A peculiarity of the Japanese network

formerly shared by AUTOVON is the use of SF trunk signaling. Failure of a trunk group will cause an instantaneous trunk processing overload, because removal of 2600 HZ signifies a call for service. With the D10 overload control, the faulty trunks are automatically locked out until the 2600 HZ returns. This prevents wasting switch resources to process faulty calls for service.

Several countries either have or are in the process of installing stored program control switches. With the increased reliability and capability of these switches, and the ease with which remote control and diagnosis can be performed, these networks are typically being installed with maintenance and operations located at a central site.

The French E10 system (Reference 22) has been in use since 1970, and a considerable amount of experience has been gained in its use. In this system, switches serving about 60,000 subscribers are grouped under a data processing center. The switches are typically unattended, and all operations and maintenance are the responsibility of the data processing center. A typical center would have a staff of about 15 network controllers, 20 people to handle subscriber services, and 20-30 maintenance technicians. These technicians are responsible for board replacement level maintenance only, as there is a single depot repair facility for the entire network. There is also an overall network supervision center, which is staffed by qualified switch and traffic engineers. It has telephone and teletype connection to all the data processing centers, but has no computational capacity of its own.

Although the E10 network has extensive centralized management, traffic controls apparently are all selected by the controllers and manually applied.

The system in the Netherlands (Reference 23) has some unique interesting features. The system consists of unmanned stored program controlled switches connected in groups of 20 to 150 switches to a central management and maintenance center. The center collects alarms and stautus information from the individual switches, and has a facility for preparing and loading control parameters into the switch memories. The center itself is only manned during normal working hours. During off hours, the center computer executes an algorithm on any alarm data to determine whether immediate maintenance is required. If it is, the computer generates a call to the maintenance man via the public radio paging system.

Other systems also have remote management and maintenance facilities of various competence levels, from remoted teletypes to processor controlled data bases. In general, the newer an installation is, the more centralized processing facility is used.

3.4 RECOMMENDATIONS FOR AUTOVON

The current AUTOVON system using 490L switches is extremely limited in its capability for higher level control, because its controls are currently primarily manual and its control program is wired into the logic. Certain system improvements already planned will increase the flexibility of the 490L system to accept controls.

The first of these improvements is the rapid access maintenance monitor (RAMM). The RAMM is currently being installed. It consists of a Data General NOVA 3D, 96K words of main memory, a moving head disk, tapes, and special interface logic to the 490L switch. The purpose of this unit is to accept the fault condition information, which previously punched a trouble card, to translate the fault condition to meaningful English, and to provide some limited troubleshooting guidance. The RAMM also provides a capability to load portions of the 490L memory from disc or tape. In order to provide these functions, a relatively general-purpose interface has been created between the RAMM and the 490L. These tasks occupy about 33% of the RAMM processor time.

The second major improvement program, called enhanced route control, will be implemented after 1980. The details of this program have not been worked out, but presumably this would involve replacement of the current fixed sequence translator and route selection subsystems with a mini or microcomputer. Common channel signalling can then be easily added, as is planned in the post-1985 time frame.

Either of these improvements could be used to provide adaptive routing for network control purposes. For the RAMM to provide this capability, interswitch telemetry circuits between RAMM's at adjacent sites would be needed and a routing update algorithm installed in each RAMM. The RAMM's could exchange trunk group status information at a regular interval and compute the proper routing tables to use given the current status. This would be a somewhat limited adaptive routing system, since the RAMM control would simply modify the routing table. No routing procedure changes can be made. Also, the RAMM would have to be told from an external source where any new trunk was connected since it could not query the trunk.

More sophisticated adaptive routing will have to wait for the computerized translator and trunk selector. With this system, trunk status information can be exchanged over the common signalling channel. If the associated mode of common channel signalling is installed, this information can include query of the identity of the distant end. If not, either a special in-band signalling mode must be used for the query or the information must come from an external source.

We recommend that some form of adaptive routing be used in the AUTOVON system as the prime item of network control. This will assure that if physical connectivity exists, the switching network will have the capability to take advantage of it. This does not eliminate the need for network visibility and control capability at higher levels, as there are situations where connectivity enhancements are required to provide adequate service.

The adaptive routing strategy should include a means of restricting traffic to primary routing only, particulary on trans-Atlantic trunks. Because of the peculiar nature of AUTOVON traffic patterns, HTR analysis as practiced by the Bell System is not needed. Trans-Atlantic destinations are always hard to reach, and very seldom is any other destination difficult.

Manual routing controls are also needed. In some cases, particularly during the initial operating period of adaptive routing, it may be desirable for the controllers to override the automatic algorithms. Also, practically speaking, the control improvements could be more easily achieved in an evolutionary manner. In the period before adaptive routing is implemented, some more flexible means of routing control is needed, and this control needs to be at a point which has visibility of the entire network.

Manual routing controls can be easily provided by the RAMM. The RAMM has the capability to reload 490L routing tables. It also has the capability to store routing tables on disk.

A program could be added to RAMM which accepts inputs from either local controllers or higher level controllers, creates a routing table, and loads it into the switch without any interruption in service. Using this technique can provide a highly flexible interim routing control until adaptive routing is developed.

Subscriber connectivity control for critical subscribers is primarily a transmission system function. In the current system, only transmission system support is provided. When a switch fails, its critical subscribers are provided access to a neighboring switch. However, their telephone number is changed by the restoral. They can originate calls but, in general, cannot receive calls since the calling parties do not know the new number. Critical subscribers need to be provided with a "roving subscriber" capability so that they retain the same number after restoral as the normal number. The 490L switch does have a code translation capability for up to 6 digits, but this is not enough to support subscriber connectivity control.

In order to properly support subscriber connecctivity control, a new code translation capability is required. While it is not practical to provide this by itself, this feature could be included in the planned routing control upgrade.

Restrictive traffic controls, as discussed in the background section, are in general undesirable, but necessary to prevent network thrashing from building up. Even though new controls, such as the queue dumping control of the D10 system, provide tighter control of network thrashing tendencies, they were not designed with a military multi-level precedence system in mind. They also would require extensive modification of the switch equipment.

The controls currently used by AUTOVON (ATOP, in conjunction with manually-controlled line load controls, trunk directionalization, and destination code cancellation) are adequate to prevent or stop network thrashing. This will be made even more true if adaptive routing is implemented since the recommended adaptive routing schema reduces the amount of ineffective signalling generated by traffic overloads. Therefore, no new traffic controls are recommended.

An improvement which is desparately needed but which has no obvious solution is some way to inform controllers that traffic controls should be applied, that a traffic overload induced network thrashing situation is imminent. Currently, this is determined by observing the ACAS display panel. This display lights a lamp when the instantaneous occupancy of a group of equipment (trunk group, or common equipment group such as RSJ) exceeds some threshold. This is an exceedingly indirect indication, requiring a controller with extensive network operating experience and a deep understanding of the underlying traffic theory for proper interpretation.

Figure 3-2 shows the basic steady state relationships for the queue which is formed by calls waiting for common equipment, as modelled by an M/M/m queue with finite waiting time and first come first served queing discipline. Traffic controls need to be applied when the probability of timeout starts to become significant, since this increases the sender holding time at the originating switch, causes more alternate routing to be used, etc. Because the network does do alternate routing, an occasional timeout is acceptable and preferable to the application of controls. As one would intuitively expect from this arguement, utilization of slightly greater than 1 is acceptable. When the utilization gets up to numbers like 1.1, it's time to apply some control.

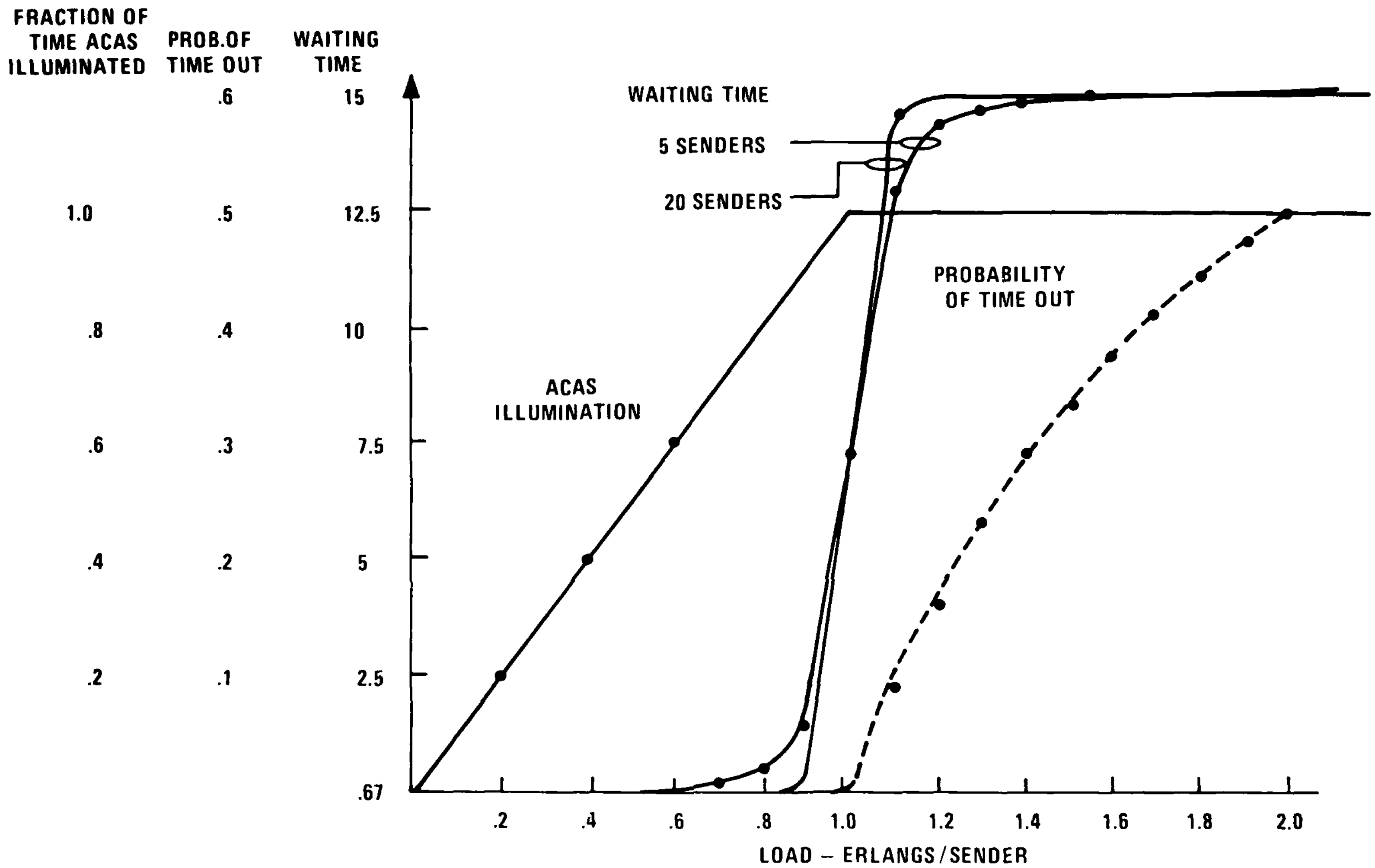


FIGURE 3-2. ACAS SWITCH CONGESTION INDICATIONS

But everywhere in this critical decision region, the ACAS display is brilliantly, solidly illuminated. To make matters even worse, these steady state percentages have to be measured over time periods of 15-30 minutes to obtain a reasonably accurate estimate, as discussed in Reference 2. The display light could easily be illuminated steadily for 5-10 minutes without indicating any need for control application. Even from these steady state considerations, it can be seen that an extremely skilled controller is needed to know when to apply traffic controls. It would be nice to directly measure the queue size and display it, or threshold it, for the controller. This is not possible with the 490L switch, and is not a practical retrofit, because of the way the marker operates. It performs scanning of trunk calls for service using a relay tree. When it finds a call for service, it siezes the internal data bus until the switch assigns a register. After a register is assigned, the marker transfers the call for service identity and then resumes scanning. If no register is available, the marker does not scan until a timeout occurs. As a result of this method of scanning, there is no direct knowledge of queue size.

The length of time waiting on queue is directly related to queue size and as shown in Figure 3-2 is a much better indicator of switch loading than is instantaneous occupancy. This parameter can be measured either by modifying TDCS or the RAMM. It is obtained by noting the time between RSJ state transitions (State 9 to State 10) and the trunk group associated with them. These times can be collected at a centralized facility, sorted by terminating switch and used to provide various indications of queue status. The simplest indicator would be to low pass filter and threshold the time for each switch, thereby providing a binary indication of switch overload. For the highly sophisticated controller, a histogram display of the past 15 minutes of activity would provide a more complete view of the switch status. Jornaling these queue time values can provide backup justification for taking control action.

The final recommendation for current traffic controls is a change in attitude. Although DCAC 310-V70-44 (reference 24) clearly states that only the minimum controls necessary to keep the network operating properly will be used, there is a natural tendency to apply too many controls too early. Then when the controller is interrogated after the fact by his superiors as to what action was taken, he can point to the control action that was taken as a demonstration that he was doing his job. The network would probably operate better if controllers were required to justify taking an action, rather than requiring them to justify not taking action. Then controllers would tend to error on the side of not taking enough controls, thereby allowing the automatic controls to operate properly.

The ultimate traffic control would be to design the network such that no controls are necessary in any condition. Although we

have been unsuccessful at determining a set of relationships which guarantees this, we believe that for some amount of processing capability and signalling bandwidth in a network of a certain number of trunks, thrashing is probably impossible. Since AUTOVON is a very small network, its route selection and signalling capabilities can be overbuilt sufficiently so that thrashing is an impossibility. Given this situation, no traffic controls are needed. The planned upgrade of AUTOVON routing capabilities and conversion to common channel signalling provide an opportunity to eliminate the need for traffic control.

REFERENCES

(1) Crowe, R.K. etal, "System Control for the Transitional DCS Technical Report #1", Honeywell Systems and Research Center, Contract No. DCA 100-78-C-0017, May 1978.

(2) Annand, F.C. etal, "System Control for the Transitional DCS Technical Report #2", Honeywell Systems and Research Center, Contract No. DCA 100-78-C-0017, December 1978.

(3) DCA European Circular 310-70-5, "European Restoral Plan for the European DCS".

(4) DCA Circular 310-70-1, "DCA Technical Control Procedures".

(5) DCA Circular 310-55-1, Vol. 1, "Operational Direction and Status Reporting in the DCS".

(6) DCA Circular 310-65-2, Vol. 1, "DCS Circuit Allocation and Engineering Manual".

(7) Morris, C.D., and F.J. Powers, "Report on Site and DCA Field Office Investigations for the System Control Improvement Program", Mitre WP-21809, Contract No. F19628-78-C-0001, July 1978.

(8) Nilsson, N., "Problem Solving Methods in Artificial Intelligence", McGraw-Hill, 1971.

(9) Sandler, G.H., "System Reliability Engineering", Prentice-Hall, 1963.

(10) "Future DCS Cost Reduction Design Factors Study", Booz-Allen Applied Research, February 28, 1978.

(11) Anon., "Analysis of Technical Control Functions", Hdqtrs USA-CEEIA Publication SCCC-TED-70-FR012 Ft. Huachuca, AZ, September 1970.

(12) Anon., "Analysis of Technical Control Functions for Croughton and Hillingdon", Computer Sciences Corp., September 1973.

(13) Anon., "Analysis of the Response Characteristics of Alternative System Control Configurations", Computer Sciences Corp., Report R4918250-1-1, Contract No. DCA 100-74-C-0034, March 1975.

(14) Franks, R.L. and R.W. Rishel, "Overload Model of Telephone Network Operation", BSTJ, Vol 52 No. 9, November 1973

(15) Bernas, G.W. and D.M. Grieco, "A Comparison of Routing Techniques for Tactical Circuit Switched Networks", ICC78 Paper 23.5.

(16) Klukis, M.K. etal, "Hybrid Computer Simulation Study of Adaptive Routing Techniques for Switching Networks to Enhance Su vivability for Optimum Network Performance", Contract No. DCA 100-74-C-0039, Martin Marietta Corp, June 1975.

(17) Burke, P.J., "Automatic Overload Controls in a Circuit Switched Communications Network", NEC68 p. 667.

(18) Laude, J.L., "Local Network Management - Overload Administration of Metropolitan Switched Networks", NEC68 p. 673.

(19) Mummert, V.S., "Network Management and its Implementation on the No. 4 ESS", ISS76 Paper 241-2.

(20) Greene, T.V. etal, "Network Management and Traffic Administration" BSTJ, Vol. 56 No.7, September 1977.

(21) Toda, A., H. Kawashima and A. Asama, "Automated Overload Control of the D10 Local Electronic Switching System", ISS76 Paper 433-1.

(22) Revel, M. and J.N. Meveur, "Maintenance and Administrative Functions in the E10 System", ISS74 Paper 442.

(23) v--- de Nieuwegiessen, D., "PMT-100, A System for Remote Controlled Management and Supervision of Telecommunication Equipment", ISS74 Paper 424.

(24) DCA Circular 310-V70-44, "Overseas AUTOVON Network Control Procedures", August 1972.

APPENDIX A - THE DATA BASE OF THE DCS NETWORK

This appendix contains the data base files that make up the DCS network data. These files are also used by the altrouting algorithm. For the most part, these files are the same as they were when they appeared in report #2. Some modifications and discussion is in order at this time because of the heavy dependancy the altrouting algorithms have of the data base format. This appendix will, therefore, not only present the modified data base file formats, but discuss the way in which the data base manipulations of the routines works.

A.1 The Connectivity Path File (CNF)

Table A-1 shows a typical file entry for this file. The modification made to this file format is the inclusion of fields which provide the mileage and transmission cost number for every link on every path.The usefulness of such entries to the cost calculation process has already been discussed. The presense of that important cost data in this file rather than the link files makes access to it much faster for the routines and allows easier access for operations personnel for review and modification.

A.2 The Station File (SF)

This file is unchanged from report #2 and is given here just for reference.

A.3 The Link File (LF)

The link file format is given in Table A-3.

The only change in this file is the addition of a field for the normal and current trunks on the link. The reason for this addtional field is to allow users of this file to be aware that another trunk has pre-empted on this link from the normally present trunk. The presense of that data here eliminates the need to scan the trunk file of the normal trunk to find out whether or not it has been pre-empted or not. An even more critical use for this data is found when restoral of the normal trunk is to be made. The trunk currently occupying this link must have some way of finding the normal user of the link when it is ready to leave. The pre-empting trunk has only its own TF to work from. If the normal trunk on the link were not listed in the LF ,then no other link to that normal user would exist. More will be said about this linking when normal trunk and circuit restoral is discussed.

A.4 The Trunk File (TF)

The format of the trunk file is given by Table A-4. A number of changes are evident to this file. The first one to mention is the

inclusion of the normal and current circuits riding the trunk. The reason for this addition is the same as the change in trunk lists in the link file. No more needs be said until restoral file manipulation is discussed later.

The listing of circuits riding the trunk is given in this file in terms of only the last four digits of the circuit's CCSD. The reason for this change is to reduce the storage required. The last four digits identify the circuit uniquely even without the first four digits.

The port types used on the circuits riding the trunk are also listed here and in the circuit files for the individual circuits. The routines need to know if patching compatibility is possible over a potential pre-emptable circuit. Since the circuit files need not be consulted during the search for any other data about the pre-emptable circuits, the inclusion of this field in the trunk file will eliminate an extra circuit file access.

The next changes in the file deal with the flags of reroute ID 1 and 2. In both cases it is necassary to let the flag indicate an additional state of the reroute circuit (that has been created by the policy of single level altroutes). The old altroute for a circuit is not unpatched until the new altroute is found. Thus, the old altroute must be identified as in place but failed an awaiting new altrouting. This delay allows unnecassry repatching of segments the old and new altroute have in common.

The circuits listed on the trunk should be listed in order of their channel numbers. This eliminates the need to access their circuit files to obtain this data.

The last change involves clarifying that the number of circuits that pre-empt segments of a circuit is more than one and fields for such lists must be expanded.

A.5 The Circuit File (CF)

The circuit file format is shown in Table A-5. The change regarding the status of the altroute is the same as for the trunk file. The field for a port type of the circuit is given for the reason mentioned earlier regarding patching compatibility.

The field for the pre-empting circuit seems to be a duplication of the possible expanded field for ID 2 and should be deleted.

A.6 The Fault File (FF)

The fault file format is presented as in Table A-6.

The fault file format originally given in Report #2 does not satisfy all of the requirements put on it by the altroute routines. For this reason, some changes have been made. The

fault report now contains a field to identify the actual segment of the trunk or circuit that has failed. The job of linking this failure to the failed trunk or circuit is now done by the "service disrupted" field. This makes only a slight increase in record size.

The use of this new structure is shown in Figure A-4. Fvery segment of failed transmission along the route of a trunk or circuit will be given a fault file record. These records are linked together by the detail pointer. Channels failed on trunks should also be given records and linked to the higher level fault files. Thus, all segments of failure and all lower levels of multiplex failure will be linked together in the fault file chain of a service.

The real needs for this segment failure identification concerns the trunk and circuit altrouting searches and the altrouting of a service previously altrouted.

The trunk altrouting search makes use of group accessible stations along a pre-emptable trunk's route. Fven though the pre-empted trunk may be failed over a link, some other link may be intact along its route and usable for an altroute. This new segment identification field of the pre-emptable trunk gives us this needed data.

A failure along an already existing altroute forces the routines to recreate the failed segment of the original service that was altrouted. The information on the original service's failures needs to be stored for this eventuality. The fault file seems to be the place for this data rather than a new file in the data base.

The circuit altroute search need not access the circuit files of possible pre-emptable circuits for any data, unless faults are to be checked on those circuits. Linking channel failures on a trunk to the trunk's fault file chain eliminates this circuit file access requirement.

TABLE A-1. CONNECTIVITY PATH FILE

Item	Comments	Bytes
Connectivity Path ID		2
Terminating Stations	Of Connectivity path, identifies the path.	6
Links and Terminating Stations (Variable - up to 12)	All of this data appears on the display.	132
Fault Pointer, Direction 1	Location, r/t, link, trunk, ckt, pointer - up to 4 such faults. Gives basic information on fault in direction 1 to be used in formatting the connectivity display and gives pointer to the fault report record.	68
Fault Pointer, Direction 2	Location, r/t, link trunk, ckt, pointer - up to 4 such faults. Same as above except for Direction 2.	68
Mileages between stations on the path. (Up to 11 links)		33
Transmission costs. (Up to 11 links)	A figure indicating cost of using each link in the path and its reliability.	22
	TOTAL	331

Number of Such Records - 25 (based on applying our definition of connectivity paths to Europe; See Figure)

Total Bytes = 25 x 331 = 8275

TABLE A-2. STATION FILE CONTENTS

Item	Comments	Bytes
Station Name		3
Station Status	Indicates if the station is totally or partly out of service or if a HAZCON exists.	1
Link ID, Status, Destination, Spare Trunks (for up to 16 links)	Identifies each link terminating at the station, its status and destination and if there are any spare trunks. Used for generating staus displays and for the operator to manually investigate alt routes.	176
Fault Detail Pointer	Points to first fault report against the station, allows that fault report to be accessed.	6
Responsible Node	Locates the station within the global data base.	3
Responsible Sector	Locates the station within the global data base.	3
Responsible Area	Loates the station within the global data base.	3
AUTODIN Site Flag	Indicates that an AUTODIN Switch is at this site, used in status display generation.	1
AUTOVON Site Flag	Indicates that an AUTOVON switch is at this site, used in status display generation.	1
ATEC Equipped Flag	Indicates that ATEC exists at this site, used to determine if communications with ATEC are possible.	1
Manned/Unmanned Flag	Indicates, if the station is a manned site, to determine what actions are possible or if there can be communications with an operator.	1
CCSD of ATEC Telemetry to Node	Permits reviewing that circuit to detemine how it can be restored or other items relative to its operational status.	8

TABLE A-2. , STATION FILE CONTENTS (Continued)

Item	Comments	Bytes
Station Reporting Status	Indicates that the telemetry to the site is out of service or that reports are overdue.	1
Time Report Due	Indicates that the time that an overdue report should have been submitted.	4
Reporting Fault Pointer	Points to first fault report relating to a telemetry failure.	6
		218

Number of Such Records - 5 stations/node x
5 nodes/sector x
4 sectors/area = 100

Total Bytes = 100 x 218 = 21,800

TABLE A-3. LINK FILE CONTENTS

Item	Comments	Bytes
Link ID		5
Terminating Stations	Stations at each end of the link, for information and for alt route sorting.	6
Trunk List (up to 16)	Trunk IDs for normal & current trunks riding this link - for impact summary, alt routing information.	192
DOD (Direction 1)	Degree of degradation (i.e., out or degraded)	1
Fault Pointer (Direction 1)	Points to first fault report, direction 1.	6
DOD (Direction 2)	Same as for Direction 1	1
Fault Pointer (Direction 2)	Same as for Direction 1	6
ETR and DTG	Estimated Time to Restore and Date/Time group when Estimate was made.	11
Highest RP	Highest restoration priority carried by the link to establish criticality of temporary/permanent restoral.	2
Connectivity Path ID		2
HAZCON		1
Data Base Distribution	List of all stations (2), nodes (2), sectors (2), and areas (2) to receive DB updates for this link. Use addressing as in ATEC 10000 Spec.	24
		257

Number of Such Records = 410*

Total Bytes = 410* 257 = 105,370

TABLE A-4. TRUNK FILE CONTENTS

Item	Contents	Bytes
Trunk ID		6
VFCT CCSD	Cross reference to VFCT identifier if this is a VFCT.	8
Link Assignments	Link number (5 bytes), terminating stations (6), super-group number (1) group number (1), type of appearance (terminating or through group)(1), assigned direction (transmit receive)(1), and whether TCF, ET, etc. (3) up to 10 links. Permits identifying carrying links to check link status and to reflect a partial outage of the link when the trunk is out.	180
CCSDs Carried	List of CCSD's carried along with the RP, channel and port type of the circuit. List both normal and current circuit. (Up to 24).	360
Reroute ID #1 and Flag	(Use 4 byte CCSD numbers). Identifies the trunk which is preplanned for restoral of this trunk, whether it is activated or if the altroute is failed and activated.	7
Reroute ID #2 and Flag	Identifies either a trunk other than the preplanned reroute which was used to restore this trunk or trunks (up to 5.) which have preempted this trunk. A flag indicates that this field is idle, or that this trunk has been rerouted or preempted or that the reroute is failed and in place.	31
Degree of Degradation (DOD), Direction 1 and Fault Location	Identifies whether entire group or partial group in direction 1 is affected, whether this is a partial degradation, out of service or a HAZCON.	4
Degree of Degradation (DOD), Direction 2 and Fault Location	Same as above, except it is for direction 2.	4

TABLE A-4. TRUNK FILE CONTENTS (Continued)

Item	Comments	Bytes
Fault Pointer, Direction 1	Points to first fault report for direction 1.	6
Fault Pointer, Direction 2	Points to first fault report for direction 2.	6
Route ID	Identifies route which this trunk rides.	5
Data Base Distribution	List of all stations (6x3), nodes (3 x 4), sectors (3 x 4), and areas (2 x 3) to receive DB updates. Use addressing as in ATEC 10000 Spec.	48
Control Responsibility		3
Networks Impacted (VON, DIN, ...)	Identifies which control functions need the data.	2
PMP		
Related Route		6
Monitoring Rgrd Flag		1
		670

Number of Such Records = 1,250*

Total Bytes = 1,250 x 670 = 837,500

TABLE A-5. CIRCUIT FILE CONTENTS

Item	Comments	Bytes
User	Identifies name of person to contact relative to this circuit.	12
Phone Number	Permits calling user.	10
RP	Restoration Priority used in impact analysis of outage.	2
VFCT Number	Identifies carrying trunk if this is a data circuit or the trunk record if this is a VFCT.	8
Trunk and Channel Number	For each segment and terminating station - up to 6. Permits identifying serving trunks for fault diagnosis, e.g., 44 JM 10 10/12, LKF, SGT.	84
Reroute ID #1 and Flag	Identifies the circuit which is preplanned for restoral of this circuit, whether it is activated and whether it has failed and activated.	9
Reroute ID #2 and Flag	Identifies either a circuit (4 byte CCSD) other than the preplanned reroute which was used to restore this circuit or other circuits (5 max) which have preempted this circuit. A flag indicates that this field is idle, or that this circuit has been rerouted or preempted or that the reroute is failed.	21
Degree of Degradation, Direction 1, and Fault Location	Identifies whether there is a complete outage or a degradation and where the fault is. Direction 1 for circuit levels faults.	4
Degree of Degradation, Direction 2, and Fault Location	Identifies whether there is a complete outage or a degradation and where the fault is. Direction 2 for circuit level faults.	4
Port Type	Identifies the type of port this circuit uses on a first-level mux.	1

TABLE A-5. CIRCUIT FILE CONTENTS (Continued)

Item	Comments	Bytes
Fault Pointer, Direction 1	Points to first fault report for direction 1.	6
Fault Pointer, Direction 2	Points to first fault report for direction 2.	6
Data Base Distribution	List all stations (6 x 3), nodes (3 x 4), sectors (3 x 4), and areas (2 x 3) to receive DB updates. Use addressing as in ATEC 10000 Spec.	48
Control Responsibility		3
		218

Number of Such Records = 10,500*

Total Bytes = 10,500 x 218 = 2,289,000

*Based on 7,400 circuits in unclassified portion of 1978 DCS connectivity data base, intra and inter Europe. This was taken to be 90% of total circuits. A 25% growth factor was added.

TABLE A-6. FAULT FILE CONTENTS

Item	Comments	Bytes
Fault ID		6
Station with Fault		3
DTG (of original report)		7
Severity	Link, group, or channel level	1
ID of link, group or channel effected	Identifies all three for channel, link and group for groups and just link for links.	9
Direction	Direction of outage	1
RFO	List of each reported, up to 3	9
ETR and DTG	The estimated time to repair and the time at which the report was made	11
DOD	Degree of degradation	1
DTG of Fault Closure		9
Station Submitting Closing Report		3
RP or Highest RP	Serviced by failed capability	2
Comments	Narrative field of fault report	80
ID of and Pointer to Fault Superceding	Identifies a fault report which superceded this fault.	12
Related Fault Pointer	Points to the first fault report related to this fault report, e.g., a transmission fault causing this AUTOVON fault.	6
Service Disturbed	List the link, trunk or circuit effected. If a whole trunk is out, do not list circuits on it. If a link is out, do not list all trunks	8

TABLE A-6. FAULT FILE CONTENTS (Continued)

Item	Comments	Bytes
Fault Detail Pointer (link, trunk or circuit)	Points to the next fault reports on this link, trunk or circuit.	6
Fault Detail Pointer	Points to the first report of a lower order which is superceded by this fault, or to the next fault which was superceded by the same fault as this report.	6
Fault Detail Pointer (to next fault report at the same station)		6
Fault Detail Pointer (to next fault report at the same node)		6
		192

Number of Such Records = 3,600*

Total Bytes = 3,600 x 192 = 691,200

A.7 The Typical File Linking

Figures A-1 and A-2 pictorially show the file linking that now exists among the data base files. The titles on the linking arrows refer to the fields in the source file that makes the link shown.

A.8 Altroute File Creation and Linking

The creation of an altroute for a circuit or trunk requires that a file in the data base be created for this new network entity. The altroute file for the circuit or trunk is the same in format as the normal data base entries. The altrouted service links to the altroute file via either the ID 1 or ID 2 fields.

The fault files that have become attached to the failed circuit or trunk file are linked to the file via the fault pointer field. This pointer is also present in the fault file to link to further faults of the trunk or circuit. The fault files link back to the trunk or circuit file they impact via the service disturbed field in each associated fault file. The removal of fault files is the key to triggering a request to the routines to remove an altroute and return to normal routing. When a fault file is removed, the link to the interrupted service is followed to find the path from the preceding fault file to the fault file following the one just removed. This path to the service is necassry to relink the fault file pointer string (see Figure A-1).

The new altroute circuit or trunk that is created now becomes the current circuit or trunk on a circuit list of a trunk file or trunk list of a link file, respectively. The altroute file also links to the trunk or link file by its own route list.

The altroute is now visible to the network via the current trunk or link pointers or the altroute pointer of the altrouted circuit or trunk.

The circuits or trunks that have been pre-empted for the altroute to exist point to that altroute via the pre-empting field of the flag 2 entry. Strings of such pre-empting pointers can also exist as the figures show.

A.9 File Manipulations in Restoral

The repair of failed equpment triggers the removal and relinking of fault files on a circuit or trunk as discussed earlier. When the last fault file is removed from a circuit or trunk file, then the algorithm is keyed into action. The service will be restored and the altroute dismantled to remove the pre-emptions that it made.

The normal route of a service can be restored if no other ciruits or trunks have pre-empted the unused portions of the normal route. This is checked by examining the pre-empting circuit or trunk field for entries.

The altroute must be removed and the services pre-empted by it repatched after the normal route is found to be restorable. The circuits or trunks that were pre-empted are found through the trunk or link assignment field of the altroute. Each link (in the case of a trunk altroute) is read and the normal trunk on that link is found. The TF for that trunk indicates whether it has been pre-empted itself and what the ultimate trunk that was pre-empted by the altroute was. (See Figure A-1). Once the pre-empted trunk is found, the trunk file of that trunk tells us the patches needed to make when the altroute patches are removed. The pointer to the current trunk riding the link is then moved from the altroute to the trunk file of the trunk pre-empted by the altroute.

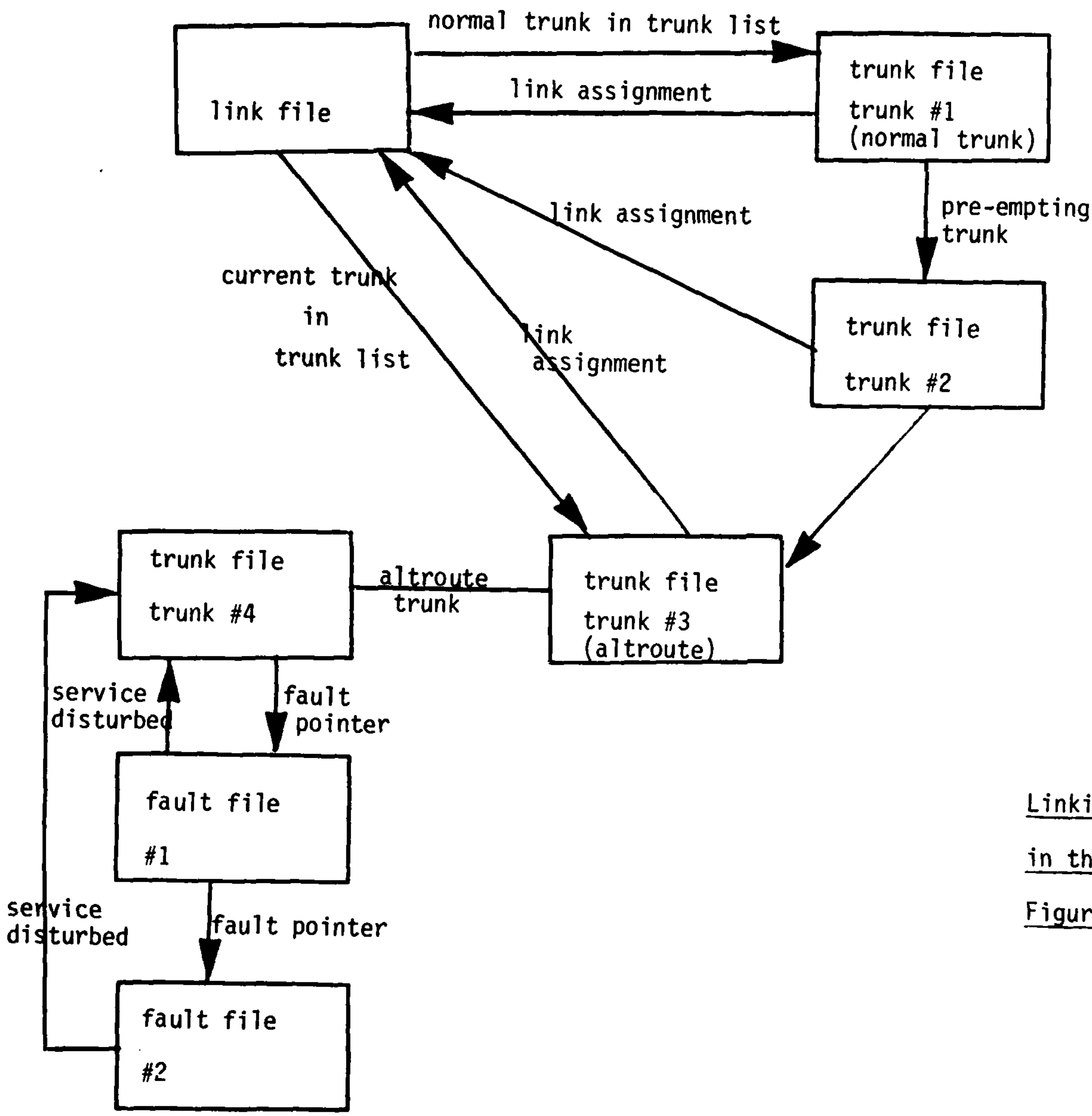
For trunks removed that have already been pre-empted out of service, the pointers of the pre-emption path are simply moved from the trunk that this altroute pre-empted to the trunk that pre-empted it. No restoral to the normal user is made in this exchange.

Of course what has been said about trunk and link files for Figure A-1 also applies for circuit and trunk files of Figure A-2 when restoral occurs.

Before concluding this discussion of restoral, one other case should be looked into. This is the case of a trunk which is pre-empted at the trunk level and then is decomposed into circuits to altroute. Although this case is probably not going to be found often ,we should explain the pre-empt linking during restoral for it. Figure A-3 shows the linking that should be made for a trunk that is decomposed into circuits. The links to the circuits showing the pre-empting trunk are needed to determine when the circuits can be restored. The reason that the trunk linking to the pre-empting trunk is not sufficient is that the next time the trunk is examined, the routine will find that it was decomposed to circuits for altrouting or restoral. Thus, the circuits will be examined rather than the trunk. The link to the circuits showing the pre-emption must be visible at the circuit level. The transfer of pre-empting trunk data to the CF's gives this visibiltiy via the linking shown in Figure A-3. The same links should also be used in the event that a circuit is decomposed into sub-vf circuits.

The restoral routine must realize that this link of pre-emptions to one lower level can occur and check the circuit files of the trunk accessed to remove pre-emptions. If pre-emptions of that trunk are found in the circuits, then they must be removed or

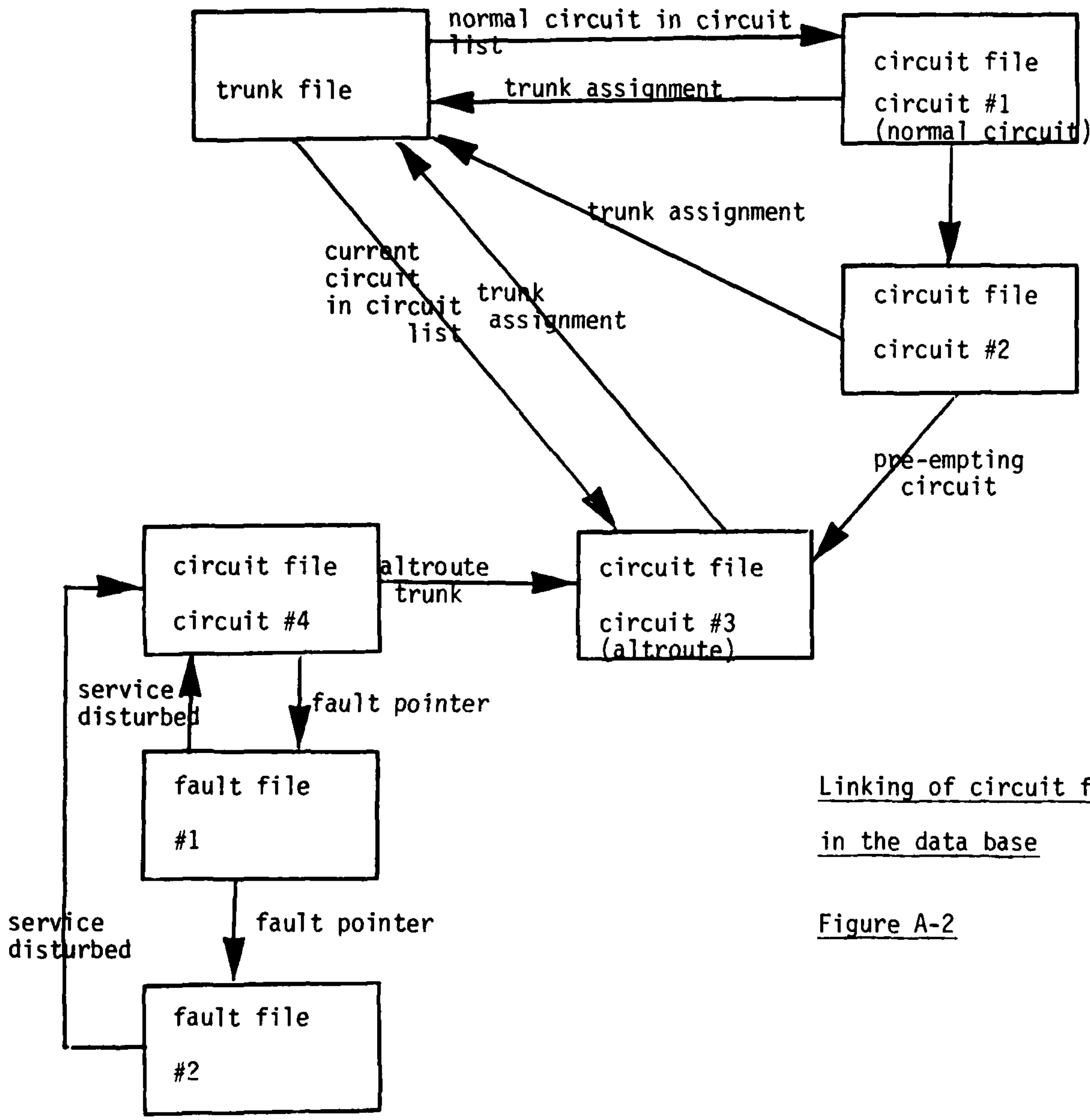
redirected. This transfer of pre-emption links will be seen only on trunks decomposed to circuits.



Linking of trunk files

in the data base

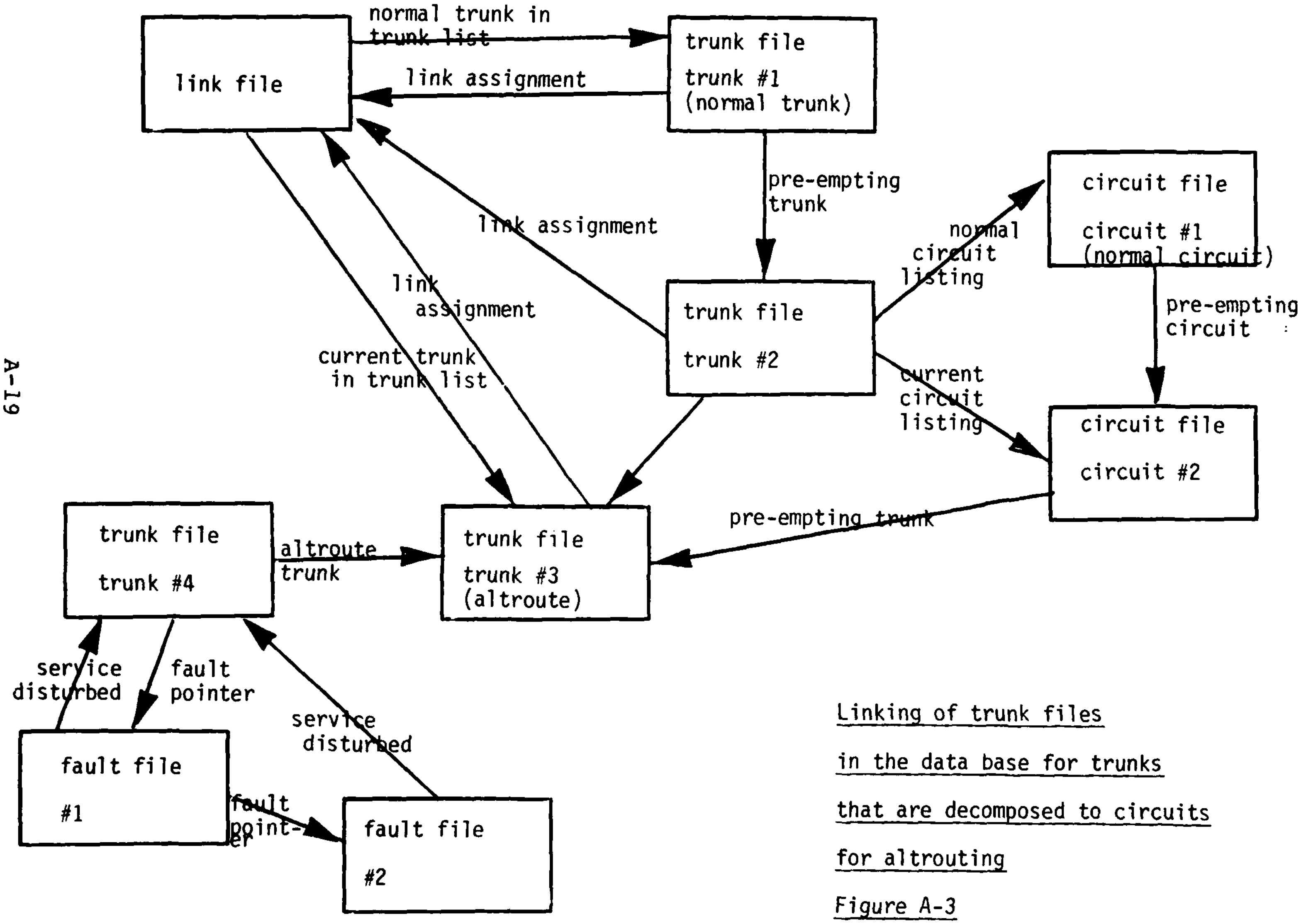
Figure A-1



Linking of circuit files

in the data base

Figure A-2



Linking of trunk files

in the data base for trunks

that are decomposed to circuits

for altrouting

Figure A-3

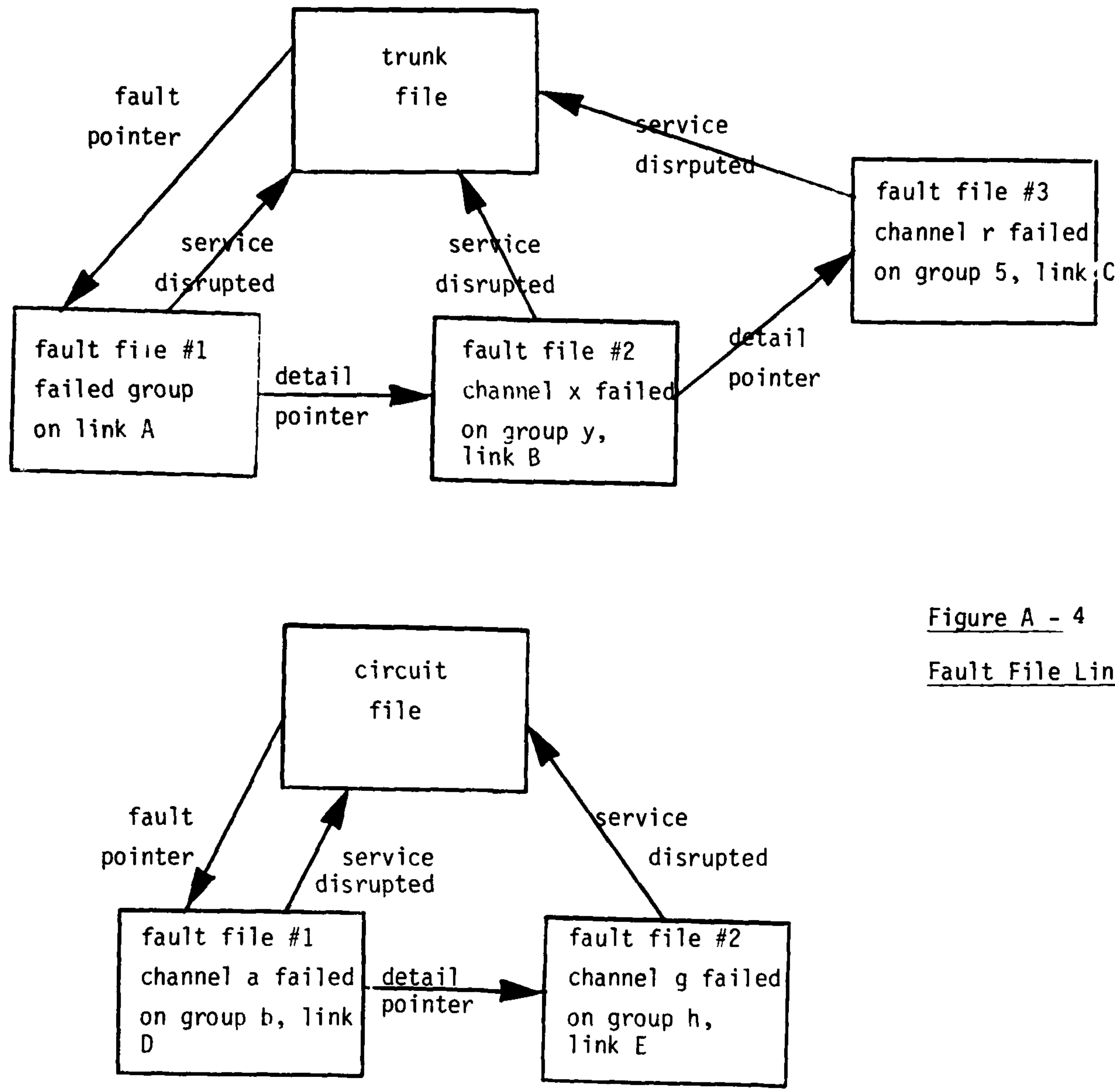


Figure A - 4

Fault File Linking

APPENDIX B-SYSTEM CONTROL MESSAGE FORMATS

Figure B-1 shows the information interfaces recommended during Task 1 of the study. The interfaces are as follows:

(1) AUTODIN NCC/DCAOC - This interface is planned as part of the AUTODIN II design to support 55-1 reporting. No changes from the AUTODIN II design were recommended.

(2) NCC/SNCC - This interface was added by the task 1 study. It forwards reports from the European packet switches to the NCC and supports status interchange between the NCC and SNCC. It uses standard AUTODIN II PSN/NCC report formats.

(3) SNCC/ACOC - This interface was added by the Task 1 study. It performs the same function as the NCC/DCAOC interface, but for the European subnetwork only.

(4) SNCC/PSN - This is a standard interface defined in the AUTODIN II network design.

(5) ATEC Sector/ACOC - This interface is basically defined in the ATEC 10,000 specification. It supports reporting of transmission system and circuit switch data. In Task 1, only the SB-3865 switchboards reported over this interface, since the TTC-39 already has substantial report transmission capability. The 490L system does not have this capability, and hence should utilize the ATEC telemetry system for reporting as was recommended for the SB-3865.

The sector/ACOC interface supports six message types related to the transmission system. These reports are as follows:

- Initial outage report
- Fault follow up report
- 55-1 daily status report
- 55-1 daily Q-line report
- Monthly performance assessment summary
- Data base update messages

These messages all follow the ATEC 10,000 format, shown in Figure B-2. The content of the initial outage and fault follow up reports are shown in Figure B-3.

(6) DSCS Nodal Control Element/ACOC - This interface is an ADCCP protocol interactive message port as specified by the DSCS Control Segment specifications. Ten message types are supported by this interface, as follows:

- o Link performance, initial report
- o Link performance, follow up
- o Equipment status, initial report
- o Equipment status, follow up
- o Link performance, summary
- o Daily calibration report
- o Alarm occurrence
- o Configuration update
- o Spare resource
- o Free text

Link performance and equipment status reports are very similar to ATEC transmission system reports, and are shown in Figures B-4 and B-5. The remaining reports are specified by the DSCS control segment design documents.

(7) DSCS Terminal Control Element/Nodal Control ELement - This interface is internal to the DSCS control segment. No change recommendations were made.

(8) ATEC Nodal Control Subsystem/Sector Control Subsystem - This interface is internal to the ATEC system. No changes in ATEC messages were recommended, but this transmission path is used for AUTOVON information reporting.

(9) TTC-39/SYSCON Channel Acquisition Unit - This interface is obsolete and no longer recommended. It was recommended to take advantage of the design characteristics of the TTC-39 switch and the TRI-TAC digital transmission group formats.

(10) SB-3865/ATEC CIS - This interface was recommended to collect SB-3865 data into system control. This same interface is an appropriate place to collect data from the 490L switches.

(11) ATEC NCS/CIS - This interface is an internal ATEC interface specified in the ATEC 10,000 specification. No changes were recommended except that it will be used as a transmission path for AUTOVON data.

(12) DSCS TCE/ATEC CIS - This interface is partially described in the DSCS control segment specifications. It is used to provide reporting of satellite status at the lowest level in the transmission system hierarchy. It supports a subset of the messages of the NCE/ACOC interface, as follows:

- o Link performance, initial report
- o Link performance, follow up
- o Equipment status, initial report
- o Equipment status, follow up
- o Free text

Link performance and equipment status formats are shown in Figures B-4 and B-5.

Details of the parameters selected and justifications are contained in reference 2.

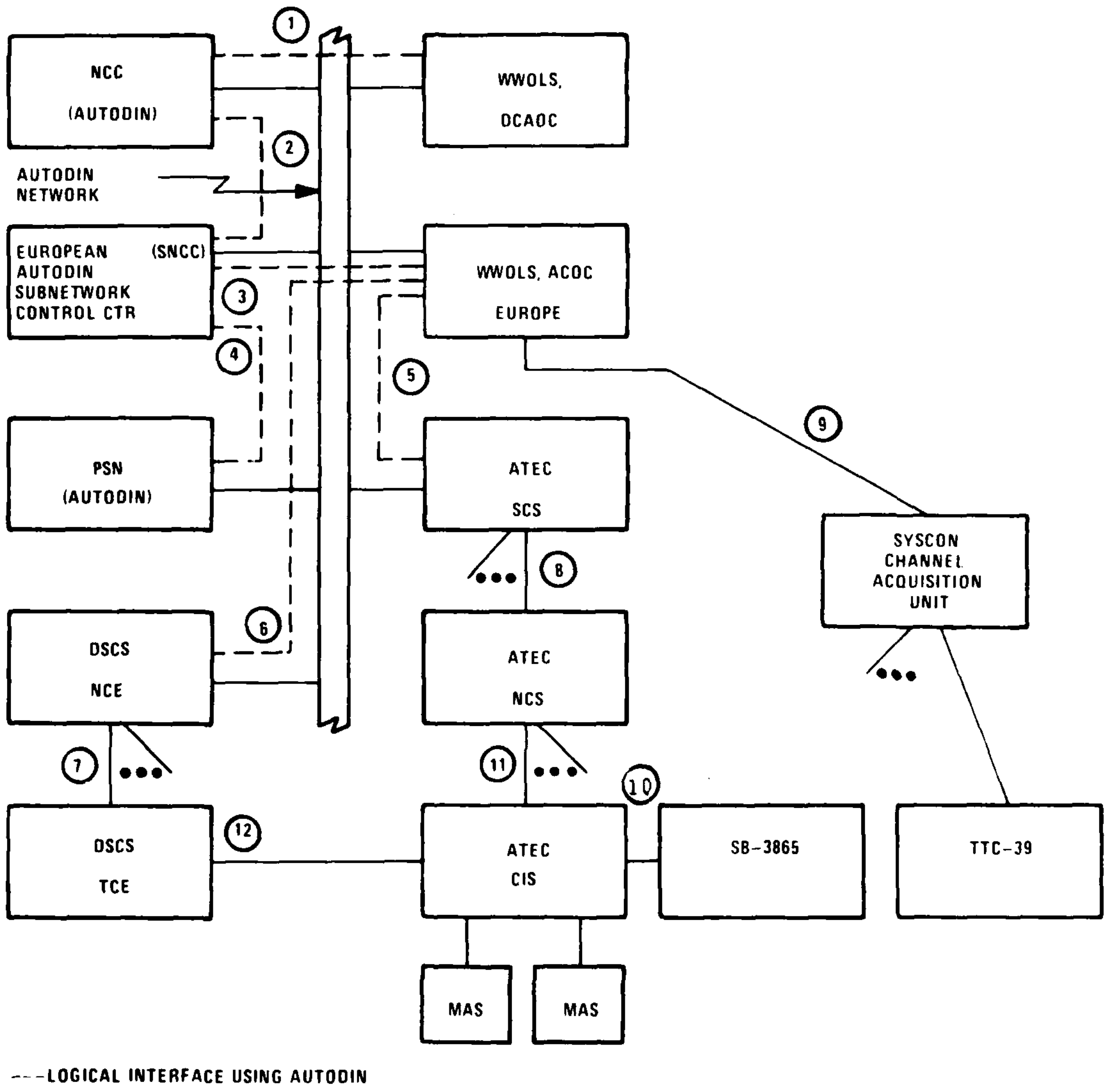
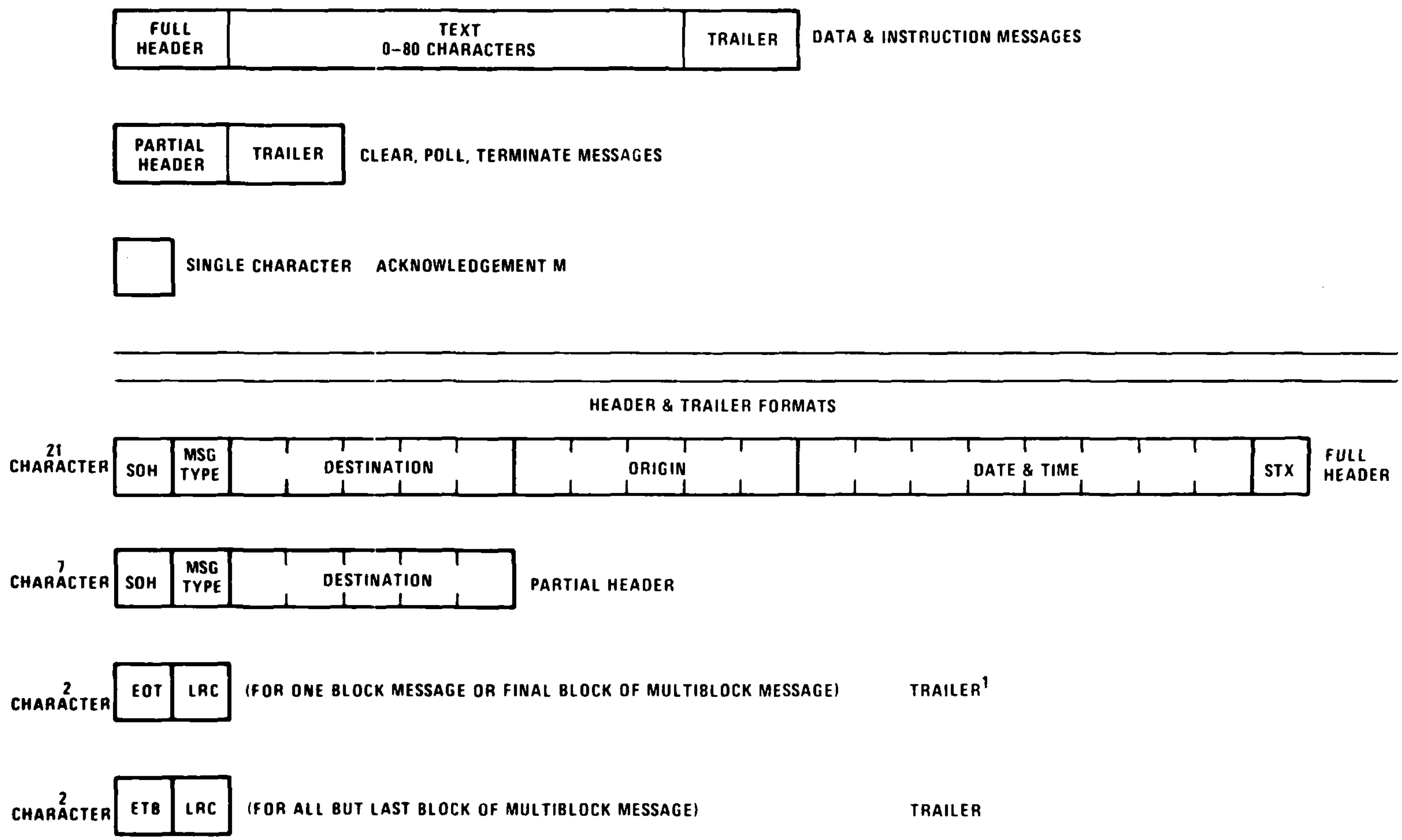


Figure B-1. SYSCON Components



NOTES

1. A "SYN" MAY COME BETWEEN EOT/EOB AND THE LRC CHARACTER
2. SUMMARIZED FROM THE ATEC 10,000 SPECIFICATION

Figure B-2. ATEC Message Formats

Initial Outage Report - 60 Characters

ITEM	Size (Characters)
Report Number (sequentially assigned module by orginating location)	4
Fault level - link, trunk or circuit	1
Circuit, trunk or link ID	8
Submitting location (usually sector)	4
Terminating locations	8
Fault location	4
Time out	8
Degree of degradation	3
Fault isolation results	20

Fault Follow-up Report - 80 Characters

ITEM	Size (Characters)
Report Number (sequentially assigned module 10,000 by originating location)	4
Previous Report Number (links this report to an initial outage report)	4
Submitting location	4
Location of fault	4
Time of this report	8
Degree of degradation	3
Estimated time of repair	4
Narrative	59

FIGURE B-3 ATEC SECTOR/ACOC REPORT FORMATS

Equipment Status Report - Initial

ITEM	Size (Characters)
Report sequence number	4
Circuit, trunk or link ID	9
Time or change	8
Submitting TCE	4
Type of equipment reported on	4
Prior status	1
Current status	1
Degree of degradation	3
Cause of outage	20

Equipment Status Report - Follow Up

ITEM	Size (Characters)
Report sequence number	4
Previous report number	4
Report time	8
Submitting TCE	4
Degree of degradation	3
Estimated restoral time	4
Narrative field	53

FIGURE B-4 DSCS RE/ATEC CIS EQUIPMENT STATUS REPORT FORMATS

Link Performance Report, Initial

ITEM	Size (Characters)
Report number	4
Link ID	8
Submitting TCE	4
Distant end TCE	4
Fault location	4
Time out	8
Degree of degradation	3
Fault details	13

Link Performance Report, Follow Up

ITEM	Size (Characters)
Report number	4
Previous report number	4
Submitting location	4
Report time	8
Degree of degradation	3
Estimated restoral time	4
Narrative field	53

FIGURE B-5 DCSC TCE/ATEC CIS LINK PERFORMANCE REPORT FORMATS

GLOSSARY

ACOC	Area Communications Operations Center
A/D	Analog/Digital
ADC	Automatic Digital Counter
ADCCP	Advanced Data Communications Control Procedure
AFSCF	Air Force Satellite Control Facility
AFSTC	Air Force Satellite Test Center, part of AFSCF
AGC	Automatic Gain Control
AN-FSC-78	Heavy Terminal
AN-GSC/24	Asynchronous Multiplexer
AN-MSC/61	Mobile Terminal
AN-USC/28	Spread Spectrum Modem
ANSI	American National Standards Institute
ASA	Automatic Specturm Analyzer
ASCT	Auxiliary Satellite Control Terminal
ATB	All Trunks Busy
ATEC	Automated Technical Control
AUTODIN	Automatic Data Interchange Network
AUTOSEVOCOM	Automatic Secure Voice Communication Network
AVOW	Analog Voice Orderwire
BC	Block Control
BER	Bit Error Rate
BIS	Baseband Interface Subsystem

BKB	Bookkeeping Block
BPSK	Biphase Shift Keying
bps	Bit per Second
CCC	Critical Control Circuit
CEI	Contract End Item
CESE	Communications Equipment Support Element
CIS	Communications Interface Subsystem (ATEC)
CIT	Controller Interface Terminal
C/kT	Ratio of carrier power to noise spectral destiny
CMC	Clear Mode Control
CODEC	Coder/Decoder
COM	Control Orderwire Master
COMSEC	Communications Security
COS	Control Orderwire Slave
COSS	Control Orderwire Subsystem
CPC	Computer Program Component
CPCI	Computer Program Configuration Item
CPU	Central Processing Unit
CS	Control Segment
CT	Control Terminal
CRT	Cathode Ray Tube
CVSD	Continuously Variable Sloped Delta Modulation
CX-11230	Cable for digital transmission groups
dB	Decibel

dBM	Decibels referenced to one milliwatt of power
DBMS	Data Base Management System
DCA	Defense Communications Agency
DCAOC	Defense Communications Agency Operation Center
DCS	Defense Communications System
DEFCON	Defense Condition
DNVT	Digital Non-Secure Voice Terminal
DRAMA	Digital Radio and Multiplex Acquisition
DSCS	Defense Satellite Communication System
DSCS/CS	Defense Satellite Communications System Control Segment
DSVT	Digital Subscriber Voice Terminal
DT&E	Development Test and Evaluation
DTG	Digital Trunk Group
DTMF	Dual Tone Multiple Frequency - An AUTOVON signalling method
DTS	Diplomatic Technical Service
EC	Earth Coverage
ECCM	Electronic Counter Counter Measure
EIA	Electronics Industries Association
EIRP	Effective Isotropic Radiated Power
EMC	Electromagnetic Compatibility
EOW	Engineering Orderwire
EPAC	Eastern Pacific Ocean (satellite network)
ESS4	Electronic Switching System #4 - Bell System Digital Tandem Switch

FDMA	Frequency Division Multiple Access
FED-STD	Federal Standard
FIFO	First In/First Out
GFE	Government Furnished Equipment
GFP	Government Furnished Property
GHz	Gigahertz
GMF	Ground Mobile Forces
G/T	Ratio of Antenna Receiving Gain to Temperature
HAZCON	Hazardous Condition - A DCS term
HIPO	Hierarchial Input Processing Output
HOL	High Order Language
HP	Historical Profile
HPA	High Power Amplifier
HSD	High Speed Data
HT	Heavy Terminal
ICD	Interface Control Drawing
ICD-004	Protocol Specification for TRI-TAC
ICF	Interconnect Facility
ICU	Interactive Control Unit (AUTODIN II)
IF	Intermediate Frequency
IMU	Inter Matrix Unit
IND	Indian Ocean (satellite network)
I/O	Input/Output
IRIS	IF-RF Interface Subsystem

JLE	Jammer Location Electronics
Kbps	Thousands bits per second
KDP	Keyboard/Display/Printer
KG	Keying Generator
KVDT	Keyboard Video Display Terminal
KY-3	Encryption Device
LANT	Atlantic Ocean
LOW	Link Orderwire
LRU	Line Replaceable Unit
MBA	Multi-Beam Antenna
Mbps	Megabits per second
MCCU	Multiple Channel Control Unit (AUTODIN II)
M_{cT}	Mean Corrective Time
M/D/1	A Markov arrival time, discrete service time, single server queue
MF2/6	Multiple Frequency, 2 out of 6 tones; siganlling method used in AUTOVON.
MHz	Megahertz
MILDEP	Military Department
MIL-STD	Military Standard
MMI	Man Machine Interface
MSF	Multiplex Signal Format
MTBF	Mean Time Between Failure
MTR	Mean Time to Restore
MTTR	Mean Time to Repair

NATO	North American Treaty Organization
NCC	Network Control Center (AUTODIN II)
NCE	Network Control Element
NCP	Network Control Processor
NCS	Nodal Control Subsystem (ATEC)
NCT	Network Control Terminal
NRE	Network Reconfiguration Engineering (AUTODIN II)
OCE	Operational Control Element
OCP	Operational Control Processor
OMS	Operation and Maintenance Subsystem
OT&E	Operational Test and Evaluation
PABX	Private Automatic Branch Exchange
PBER	Pseudo Bit Error Rate
PBX	Private Branch Exchange
PCM	Pulse Code Modulation
PM	Performance Monitoring
PMP	Performance Monitoring Program
PN	Pseudo Noise
PN/TDMA	Pseudo Noise/Time Division Multiple Access
PSN	Packet Switching Node
PTF	Patch and Test Facility
QPSK	Quadraphase Shift Keying
RF	Radio Frequency
RFO	Reason for Outage

RSJ	Register Sender Junctor
RSL	Received Signal Level
RSS	Received Signal Strength
RTS	Remote Tracking Station, associated with the AFSCF
SATCOM	Satellite Communications
SB-3865	The user concentrating switch for upgraded AUTOVON/ AUTOSEVOCOM II.
SCAU	SYSCON Channel Acquisition Unit
SCCE	Satellite Configuration Control Element
SCCU	Single Channel Control Unit
SCM	Switch Control Module
SCR	Silicon Controlled Rectifier
SCS	Sector Control Subsystem (ATEC)
SDMX	Space Division Matrix
SF	Single Frequency
SMS	Satellite Monitoring Subsystem
SNCC	Subnetwork Control Center; a copy of the NCC serving European AUTODIN only.
SNCE	Subnet Control Element
SSMA	Spread Spectrum Multiple Access
SSME	Spread Specturm Modem Equipment
STED	Seeley Trunk Encryption Device (SB-3865)
SYSCON	System Control
TAC	Terminal Access Controller
TBD	To be determined

TBP	To be provided
TBS	To be specified
TCC	Transmission Control Code (AUTODIN II)
TCCF	Tactical Communications Control Facility
TCE	Terminal Control Element
TCP	Terminal Control Processor
TDM	Time Division Multiplex
TDMA	Time Division Multiplex Access
TDMX	Time Division Matrix
TLC	Traffic Load Control
TM	Test Mode
TRI-TAC	The tactical communications system currently under development.
TTC-39	The nodal circuit switch for upgraded AUTOVON/AUTOSEVOCOM II.
TTY	Teletype
TX	Transmission
UK	United Kingdom
ULS	Unit Level Switch
VDCU	Voice Data Channel Unit
VOW	Voice Orderwire
WF-16	Fieldwire for telephone installations.
WOFR	WWOLS Output Formatting Routines
WPAC	Western Pacific Ocean (satellite network)
WWOLS	World Wide On-Line System